Exhibit A

Mueller's Protected Content in the Public Domain

*Product Bulletin*

# MSP430 Ultra-low-power Microcontrollers

Second Quarter, 1999

**MSP430 Key Features**
- Ultra-low power consumption
  - 400-µA active mode
  - 1.3-µA standby mode
  - 0.1-µA off mode
- High throughput processor
  - 16-bit orthogonal RISC architecture
  - Most instructions executed within a single 200-ns cycle operating at 5 MHz
  - Seven different address modes for 51 (27 core) instructions
- Hardware multiplier
- Integrated 14-bit A/D converter
- Integrated LCD driver
- Integrated USART
- Various timers

From the beginning, the design objective of the MSP430 team was to focus on the ultra-low power consumption of the complete system. The goal was to create a microcontroller which consumes very little current in the sleep modes and performs the given tasks in the active mode as fast as possible.

To reduce the current consumption of a system, the MSP430 allows designers the ability to influence the active current consumption and active time as well as sleep mode current consumption and sleep time.

The **active mode** current consumption of the MSP430 is 400 µA in a typical 3-V system. The time to **wake-up** from the sleep mode to a total functional system takes a maximum of **6 µs**. This allows the MSP430 to be in sleep modes longer and eliminates unnecessary energy use in the active mode. The powerful 16-bit CPU core ensures a **fast execution of the tasks** and therefore reduces the active time. This means that the higher the performance of the CPU core, the lower the system power


*A full range of MSP430 evaluation and support tools are available and provide easy-to-use design solutions.*

### *The 'Green' Microcontroller*

In a modern household, many electronic applications like TV sets or stereo systems are permanently in a standby mode. Assuming the total standby power in a household is 10 W, a country with 40 million households requires 400 MW just to supply the standby energy. This means that a mid sized power plant is working only to supply the standby energy for these parts.

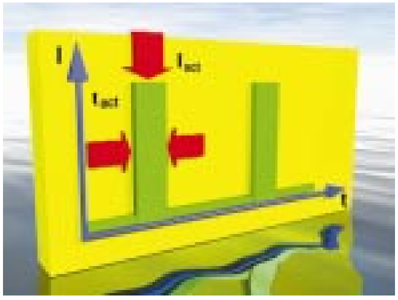
The MSP430 is an ultra-low-power microcontroller family and can help to reduce this standby current. The typical current consumption in low-power mode is 1.3 µA, where the device is still capable of displaying information on the LCD display or keeping a real-time clock updated.

This ultra-low-power consumption is no limitation for the outstanding high processing capability. The 16-bit RISC CPU core can perform tasks like calculation of the energy, faster than conventional 4- and 8-bit microcontrollers. This combination sets a new benchmark of processing capability versus energy consumption.

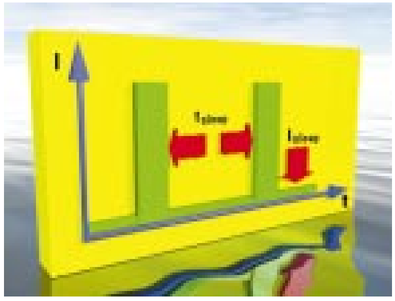
The MSP430 offers 1200 MIPS/Watt in active mode. Finally, the high integration of the MSP430 allows the user to build up a system with a minimum of external components. This leads to very cost competitive system solutions.

consumption. All MSP430 peripheral modules are specially designed to support these ultra-low power features.

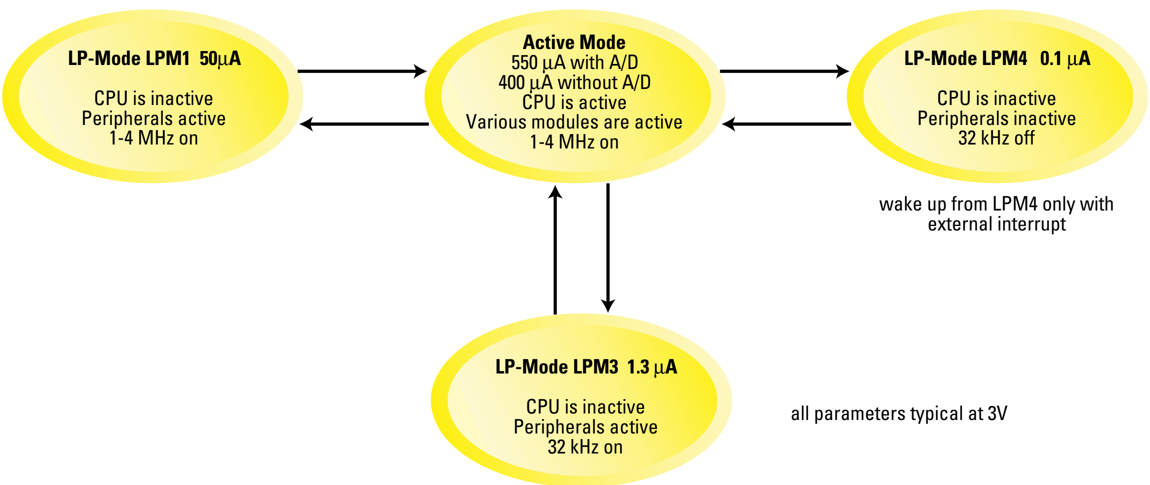
The **sleep modes** offer a reduced current consumption even when some peripherals are still active. For example, in a simple real time clock (RTC), it is **not** necessary to keep the device in active mode. Another example, the system can operate from the 32-kHz (ACLK) clock instead of 1-4 MHz (MCLK) with the timers and LCD still active. These examples are benefits of the most often used low-power mode 3 (LPM3) which consumes 1.3 µA typically. The current consumption can be reduced down to 0.1 µA in LPM4 where the MSP430 is still capable of processing external interrupts, for example from a connected keyboard. The sleep time can be maximized due to the fast wake-up from the low-power modes.



*Current consumption in active mode*



*Current consumption in sleep mode*




### *System Cost Saving with the MSP430 Family*

The MSP430 offers a variety of possibilities to reduce the cost of the complete system to a minimum.

- The use of the 32-kHz XTAL and the internal DCO/FLL eliminates the need for a second XTAL for the system frequency. Furthermore, a ceramic resonator can be used in place of the 32-kHz XTAL; or, the system can be operated without any external component for the clock generation at all.
- The low-power features of the MSP430 make it possible to choose a smaller battery for the application and still increase the system life due to the various power saving modes.
- High code efficiency leads to smaller memory sizes and drives cost down.
- The high integration of the device makes an external LCD driver or an external ADC unnecessary. This high level of integration saves system cost and lowers the failure rate of the system by reducing the device count.
- The ease-of-use MSP430 architecture and the development tools significantly improve the development time and speed up the time-to-market.

## The MSP430 RISC Core

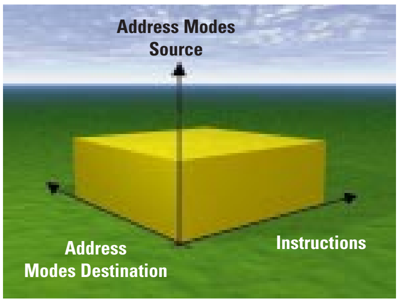
***16-bit RISC CPU***

The MSP430 CPU offers you much more than standard 4- and 8-bit microcontrollers. The 16-bit RISC core is built with a highly orthogonal structure.

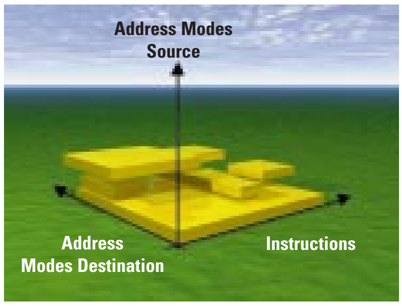
Every instruction can be used with each of the seven different addressing modes. The reduced instruction set consists of only 27 core instructions. However, the user has, due to the 24 additional emulated instructions the capability of using highly familiar instructions. For example, a familiar instruction like INC R4 is available to the firmware programmer and automatically emulated by the assembler with ADD #1,R4. This instruction will be executed like all other register to register instructions in only one cycle.

The orthogonal architecture of the MSP430 CPU core makes the device extremely easy-to-use.

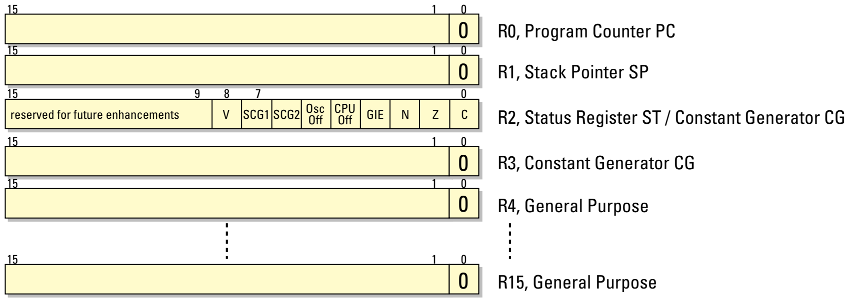
Sixteen (16) registers are implemented in the CPU itself and contain the Program counter, the stack pointer, the Status Register and the Constant Generator (which contains the highly used constants -1, 0, 1, 2, 4 and 8). This feature makes the MSP430 an extremely code efficient device using a lot less of the code space than conventional CISC machines. The remaining 12 registers are available for general usage.




*Example: orthogonality to operand instructions*




*Example: non-orthogonality to operand instructions*




The hardware multiplication module in the MSP430x33x configuration provides multiplications in less than one cycle. The two operands are moved into registers inside the multiplier module, and in the next cycle, the result can be read out.

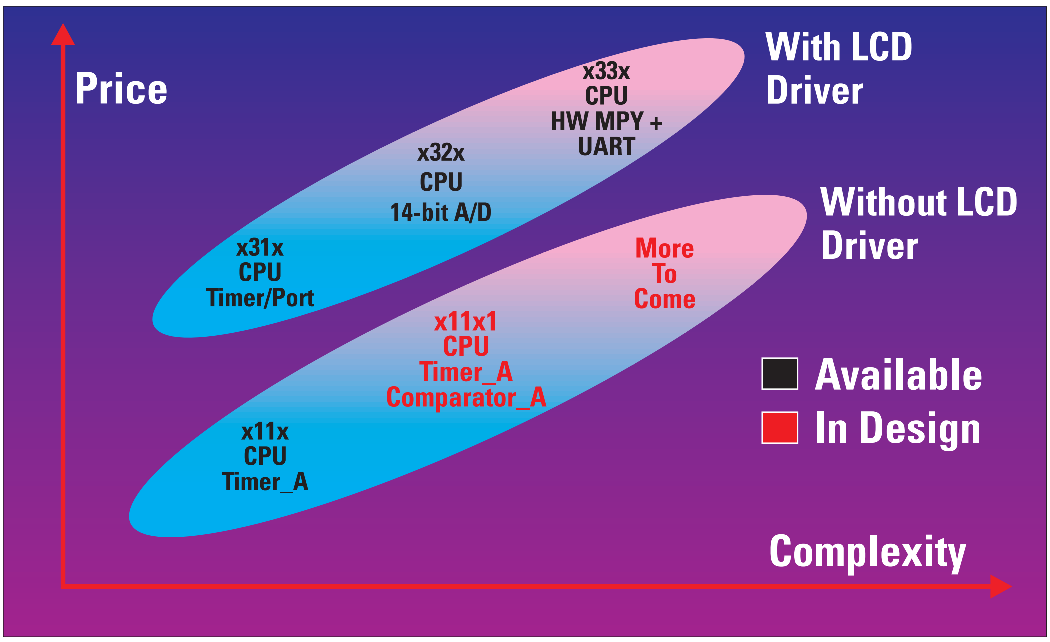
***Oscillator / FLL Module***

The clock network of the MSP430 offers the designer flexibility. The Digital Controlled Oscillator (DCO) generates the system frequency of 1 to 4 MHz. The 32-kHz oscillator, which operates with only a single 32-kHz crystal, can be used to provide a stable frequency over temperature and operating voltage. The integrated Frequency Locked Loop (FLL) regulates the system frequency MCLK with the stable 32-kHz crystal frequency. It is even possible to operate the MSP430 without any crystal at all, disable the FLL and just use the DCO to generate the system clock.

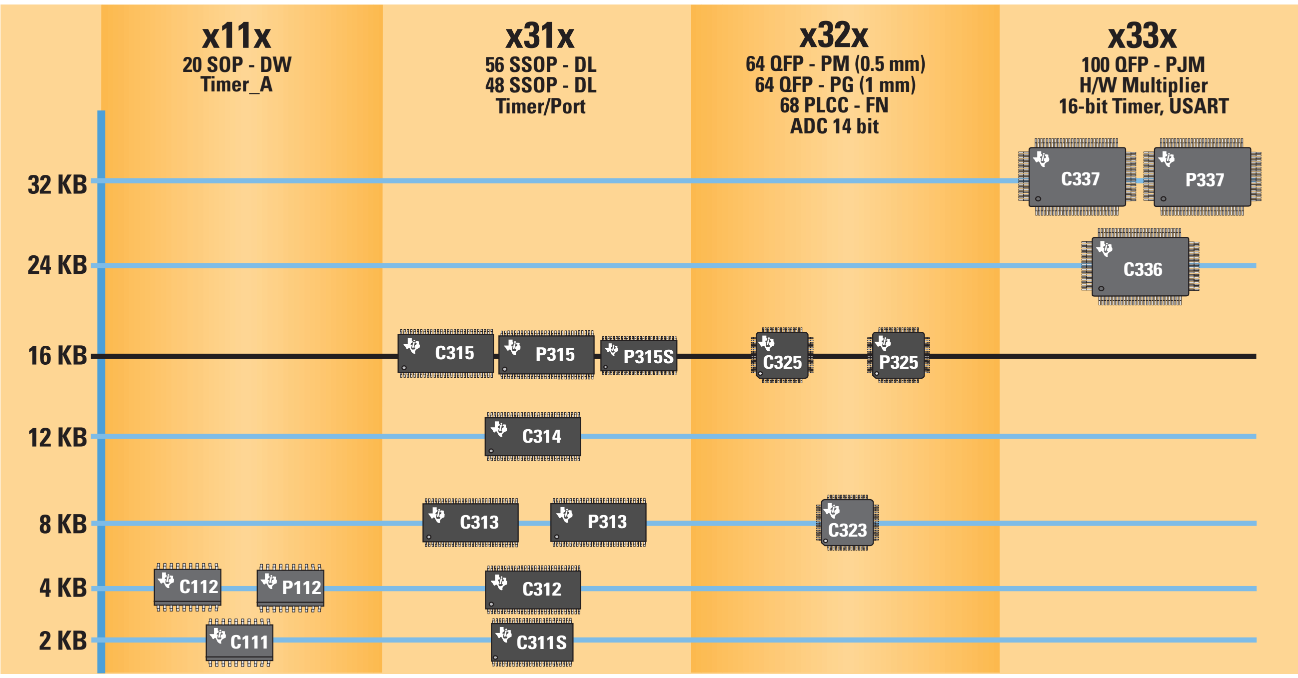
The product offers a fail-safe feature. If the crystal connection is broken, the MSP430 continues operation with the lowest possible frequency.

The DCO starts operation a maximum of 6 µs after a reset or interrupt occurs. This provides a working system in a fraction of the time needed with conventional microcontrollers.

## The MSP430 Mixed-Signal Processor Roadmap



## The MSP430 Mixed-Signal Processor Family




The MSP430 family offers you flexibility. Choose from several members out of four configurations:

- MSP430x11x devices – Basic version with Timer_A
- MSP430x31x devices – A/D conversion with the Timer/Port module
- MSP430x32x devices – A/D conversion with 14-bit ADC module
- MSP430x33x devices – High-end version with Timer_A, USART and H/W multiplier

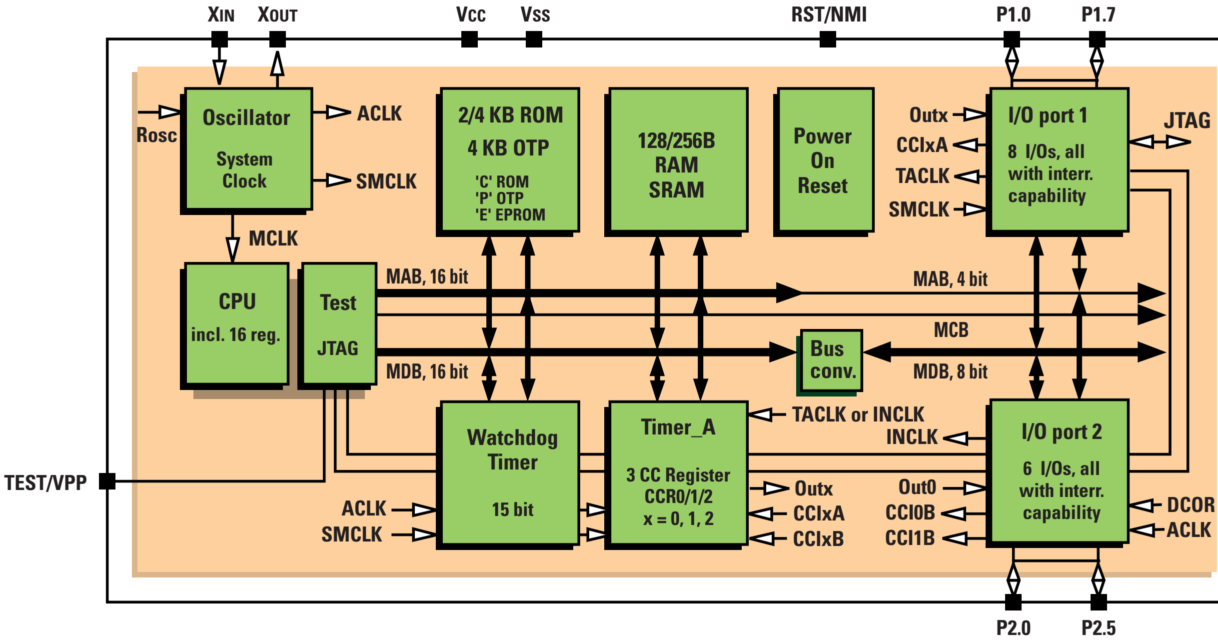
## MSP430 Mixed-Signal Processor Selection Guide

| DEVICE | ROM | OTP | EPROM | RAM | ADC | LCD^ | PERIPHERALS | PACKAGE |
|---|---|---|---|---|---|---|---|---|
| MSP430C111 | 2KB | | | 128B | slope | N/A | WDT, P1, P2, T_A | 20 SOP |
| MSP430C112 | 4KB | | | 256B | slope | N/A | WDT, P1, P2, T_A | 20 SOP |
| MSP430P112 | | 4KB | | 256B | slope | N/A | WDT, P1, P2, T_A | 20 SOP |
| PMS430E112 | | | 4KB | 256B | slope | N/A | WDT, P1, P2, T_A | 20 DIL |
| MSP430C311S | 2KB | | | 128B | slope | 64seg | WDT, BT, T/P, P0, 8bT/C | 48 SSOP |
| MSP430C312 | 4KB | | | 256B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 56 SSOP |
| MSP430C313 | 8KB | | | 256B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 56 SSOP |
| MSP430C314 | 12KB | | | 512B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 56 SSOP |
| MSP430C315 | 16KB | | | 512B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 56 SSOP |
| MSP430P313 | | 8KB | | 256B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 56 SSOP |
| PMS430E313 | | | 8KB | 256B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 68 CLCC |
| MSP430P315 | | 16KB | | 512B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 56 SSOP |
| MSP430P315S | | 16KB | | 512B | slope | 64seg | WDT, BT, T/P, P0, 8bT/C | 48 SSOP |
| PMS430E315 | | | 16KB | 512B | slope | 92seg | WDT, BT, T/P, P0, 8bT/C | 68 CLCC |
| MSP430C323 | 8KB | | | 256B | 14 bit & slope | 84seg | WDT, BT, T/P, P0, 8bT/C | 64 QFP/68 PLCC |
| MSP430C325 | 16KB | | | 512B | 14 bit & slope | 84seg | WDT, BT, T/P, P0, 8bT/C | 64 QFP/68 PLCC |
| MSP430P325 | | 16KB | | 512B | 14 bit & slope | 84seg | WDT, BT, T/P, P0, 8bT/C | 64 QFP/68 PLCC |
| PMS430E325 | | | 16KB | 512B | 14 bit & slope | 84seg | WDT, BT, T/P, P0, 8bT/C | 68 CLCC |
| MSP430C336 | 24KB | | | 1KB | slope | 120seg | 31x + T_A, MPY, USART, P1, P2, P3, P4 | 100 QFP |
| MSP430C337 | 32KB | | | 1KB | slope | 120seg | 31x + T_A, MPY, USART, P1, P2, P3, P4 | 100 QFP |
| MSP430P337 | | 32KB | | 1KB | slope | 120seg | 31x + T_A, MPY, USART, P1, P2, P3, P4 | 100 QFP |
| PMS430E337 | | | 32KB | 1KB | slope | 120seg | 31x + T_A, MPY, USART, P1, P2, P3, P4 | 100 CQFP |

**C=ROM, P=OTP, E=UV EPROM (prototyping only)**

ADC: slope = slope measurements for resistive sensors, 14-bit = hardwired A/D converter with 14-bit resolution.
LCD = LCD Driver for 1-4 MUX, WDT = Watchdog Timer, BT = Basic Timer, T/P = Timer/Port Module, 8bT/C = 8-bit Timer/Counter, T_A = 16-bit Timer, MPY = Hardware Multiplier, USART = UART/SPI, P0 – P2 = I/O Ports.
^ If not used for LCD, the LCD segment lines can be used as outputs.

## MSP430x11x Configuration




The newest member in the MSP430 microcontroller family offers an unmatched ratio of ultra-low power consumption, 16-bit RISC performance, and low cost. The MSP430x11x family features an ultra-low power consumption rating of 350 µA in active mode, 1.5 µA in standby mode and 0.1 µA in (RAM-retention) off mode. With volume pricing as low as $1 for ROM versions, the 11x devices are quickly becoming the benchmark in this price/performance range.

The 16-bit RISC core operates up to 5 MHz, allowing most of the 27 core instructions to be executed within one 200-ns cycle. This combination of 27 core instructions and the orthogonal architecture allows every instruction to be used in each addressing mode, making the MSP430x11x easy to program. It's also completely code compatible with TI's other MSP430 families.

Among the peripherals integrated into the new device are: a 16-bit timer with three capture/compare, 14 individual I/O signals, a new basic clock system, watchdog timer and JTAG interface. The new basic clock system makes it possible to increase the maximum standby time to 128 seconds. The memory sizes are 2 or 4 KB with the ROM version and 4 KB with the low-power OTP version. The OTP version (MSP430P112IDW) is available now.

### *Typical applications*

The MSP430x11x is ideally suited for portable instrumentation equipment such as:

- Real-time clock
- Communication
- Digital motor control
- Home automation
- Alarm systems
- Data loggers

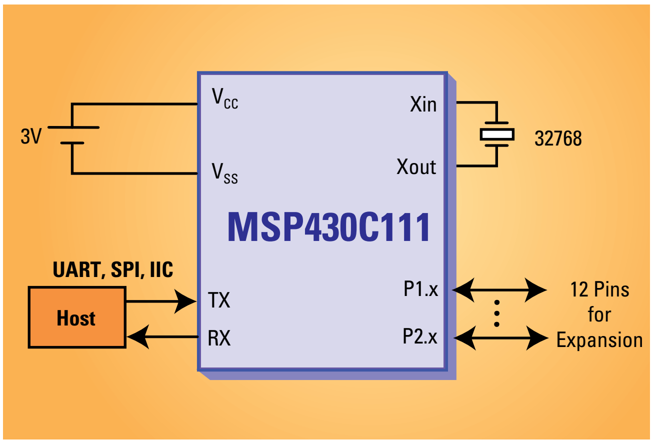
### *Development tools*

Development tools for the MSP430x11x include an Evaluation Kit from TI, and in-circuit emulators and C-Compilers are available from third-party development tool vendors.

## MSP430x11x Application Example

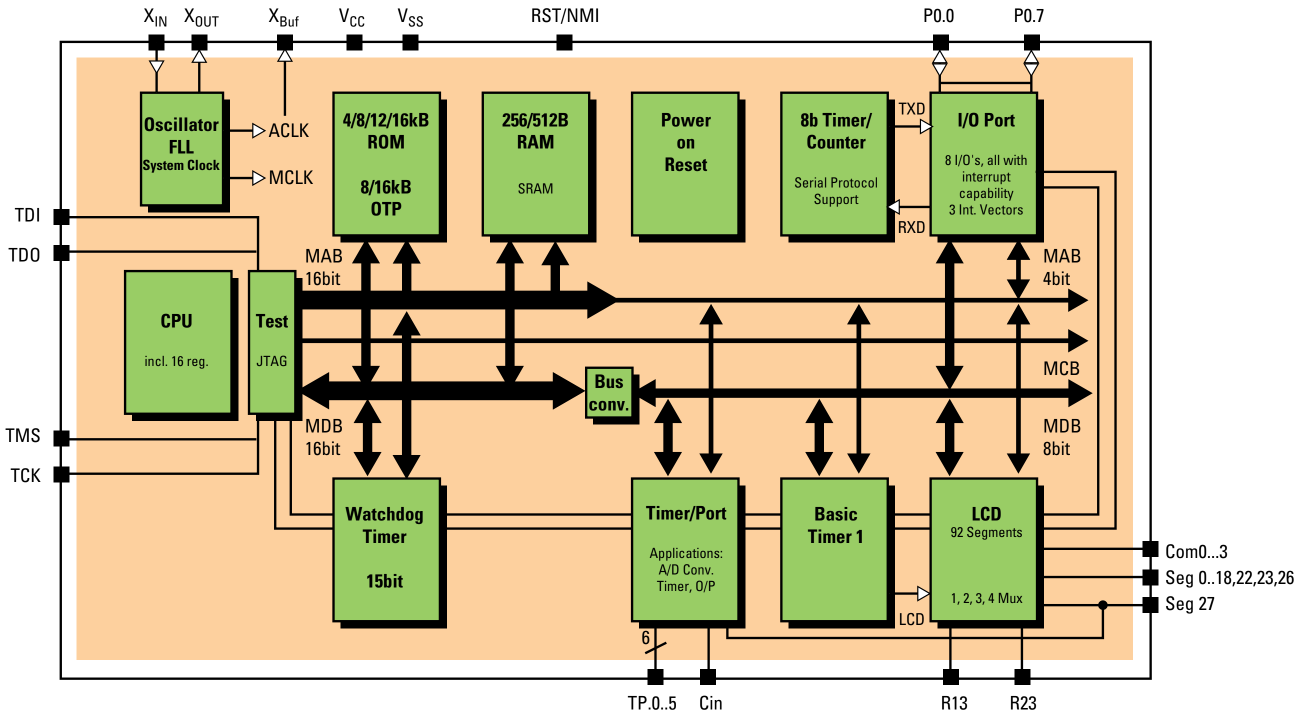
The MSP430x11x series, new members to the MSP430 family, expands the reach of low-cost system design to a higher level. A 10-year Real-Time Clock is one of many system applications that minimize the cost to manufacturers while simplify the design and development cycle drastically.

Working at Low Power Mode 3 (LPM3), the MSP430x11x receives the time input from a 32,768-Hz external crystal oscillator. With a connection to an outside host device through UART or SPI port, MSP430x11x internally generates a timer interrupt service routine to wake-up once per second. System integration engineers will be able to develop an application specific system by taking advantage of this 1.52-µA average current consumption solution.




*Low-cost 10-year Real-Time Clock (RTC)*

## MSP430x31x Configuration




The MSP430x31x configuration is the most cost optimized version with the LCD driver in the MSP430 family. It offers all the family features like **16-bit RISC core** and **ultra-low power** consumption.

It can be used for sensor applications by measuring the resistor values. The **Timer/Port module** can perform this resistive to digital conversion by measuring the charge/discharge time of an external capacitor. If this function is not needed, this module can be used as a 16-bit Timer.

The **Watchdog Timer** can be used in 15-bit watchdog mode or in 16-bit general-purpose timer mode.

The **Basic Timer** includes two 8-bit timers for general use. It generates the basic LCD frequency and supports the real time clock function.

The **LCD driver** module can drive up to 92 segments in 1 - 4 MUX mode in this configuration.

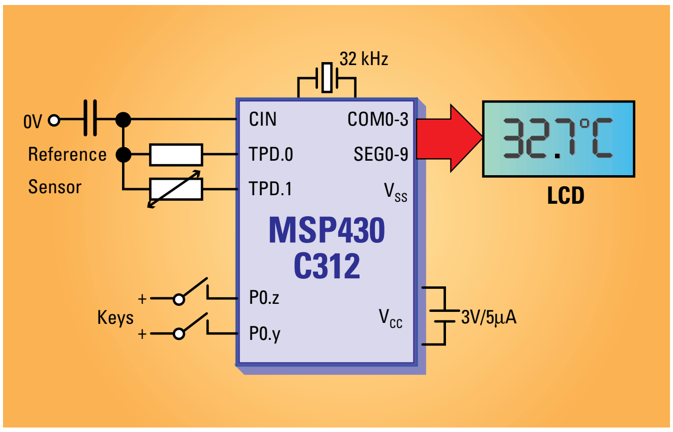
The **I/O port** can be individually configured and each pin has interrupt capability. The **8-bit Timer/Counter** supports serial communication protocols like UART or I²C bus on the I/O port, the software routines are included in the MSP430 application report.

## MSP430x31x Application Example

The MSP430 requires only a minimum of external components to build up a complete system.

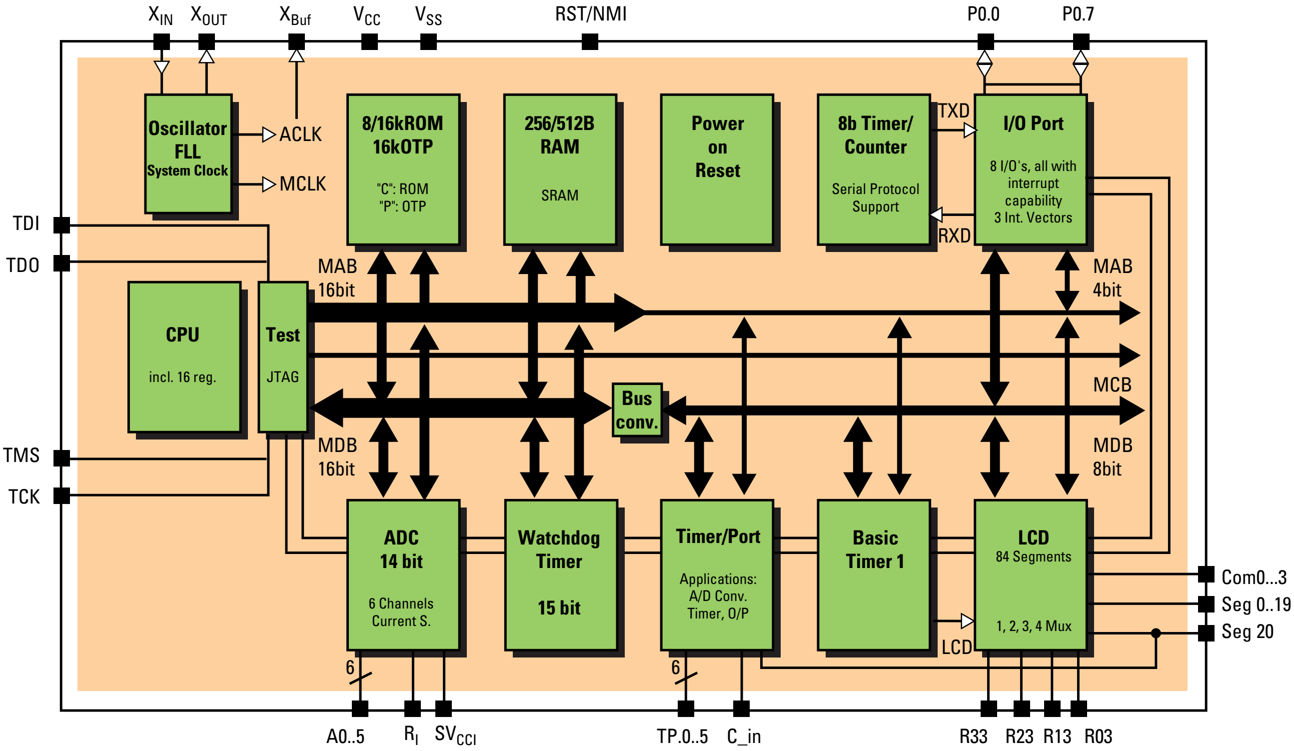
The low component count in the thermometer shown ensures a fast design cycle, maximum flexibility via software and a very competitive system cost.

For resistive sensors (such as thermistors) the 16-bit A/D converter integrated in the Timer/Port Module is ideal. Three (3) additional external components: a crystal (optional), a battery and the LCD display are all that you need for a complete system with the MSP430.




*Thermometer application*

## MSP430x32x Configuration

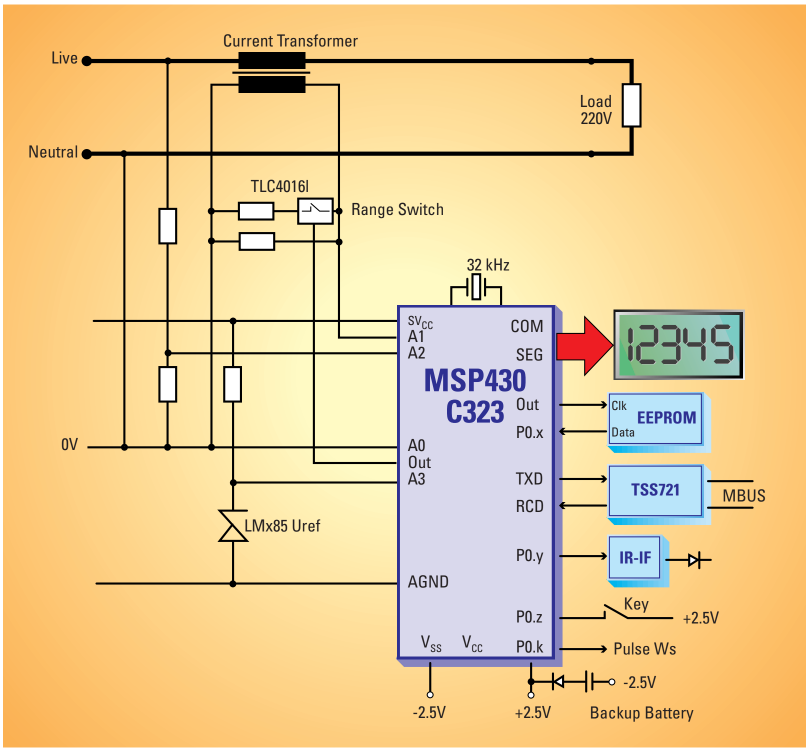



The second configuration in the MSP430 family, the MSP430x32x, offers a high resolution A/D converter in addition to the peripherals of the MSP430x31x.

This **14-bit A/D converter** has six inputs to convert analog signals into a 14-bit digital value over the full supply voltage range; or a 12-bit resolution in each of four separate ranges. The integrated current source can be programmed with an external resistor to connect current driven sensors.

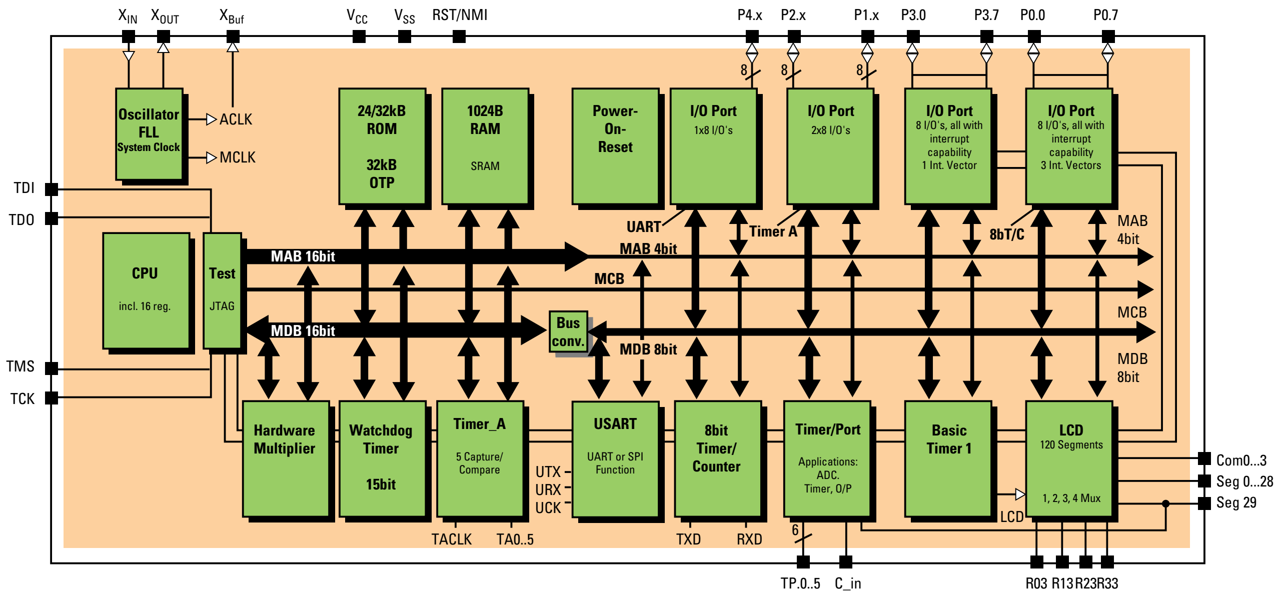
## MSP430x32x Application Example

One of the many application examples in the MSP430 application report is the shown single-phase electricity meter with the MSP430. This cost competitive single-chip solution uses the 14-bit ADC to convert the analog signals of the voltage (via a voltage divider on A2) and the current (via a current transformer on A1) into a digital value. A split power supply enables signed measurement, while the LMx85 creates a reference voltage on A3. The value of the consumed energy is calculated by the CPU and directly displayed on the LCD or transmitted via the optional serial (MBUS) connection. Non-volatile data storage is possible with an optional external EEPROM.




*Single phase electricity meter*

## MSP430x33x Configuration




The new MSP430x33x configuration is focused on the high-end area of applications. For this reason, this device generation was equipped with some additional features.

The **H/W multiplier** module performs 16x16, 16x8, 8x16 and 8x8 multiplications with signed, unsigned and unsigned accumulation. After loading both operands into the multiplier, the result is available in a separate register; no extra cycle is needed for the multiplication.

The Timer_A module includes a 16-bit timer/counter and five capture / compare registers, which can be configured by the application to capture or compare mode. The capture mode is primarily used to measure external or internal events from any combination of positive, negative or both edges; but, can also be stopped by software. The compare mode is primarily used to generate timing for the software or application hardware or to generate pulse-width modulated output signals for various purposes like D/A conversion functions, PWM, or motor control.

The **USART** module has two functions for serial communication included, a standard asynchronous communication protocol (UART up to 115.2 kBaud) and a serial peripheral interface function (SPI). One bit in a control register defines the mode (which can be switched back and forth in the application).

The number of **I/O pins** is increased in this configuration to 40. The **LCD** offers 30 segment lines to drive up to 120 segments in 4MUX mode.

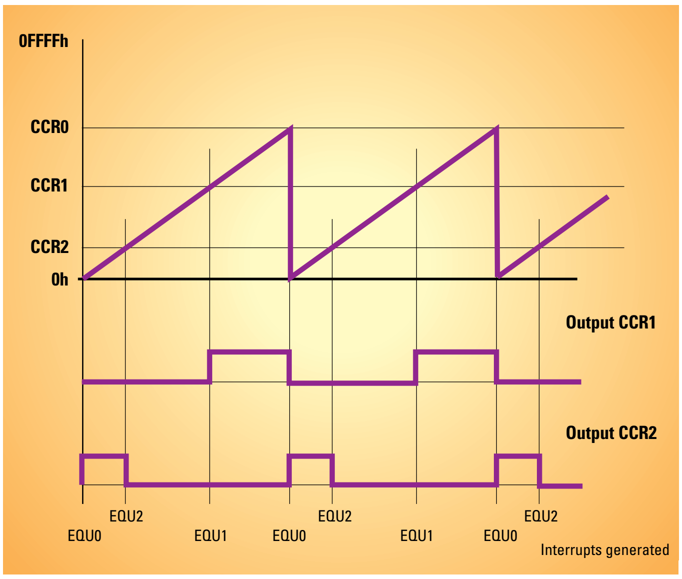
## MSP430x33x Application Example

The Timer_A of the MSP430x33x configuration is ideally suited for the generation of PWM-signals in several modes.

**Continuous Mode:** the Timer Register runs continuously upwards and continues at zero after the value 0FFFFh. This mode allows up to five independent timing.

**Up Mode:** the Timer Register counts up to the content of Compare Register 0 (here the Period Register) and restarts at zero when it has reached this value. If the Timer Register equals one of the four Compare Latches the hardware task programmed to the Output Unit is performed.

**Up-Down Mode:** the Timer Register counts up to the content of Capture/Compare Register 0 (here the Period Register) and counts down to zero when it has reached this value.




*PWM with the Timer_A (motor control application)*

## The MSP430 Starter Kit

***MSP-STK430x320—a Low-cost Plug-and-Play Tool***

The Starter Kit (STK) enables you to explore the MSP430 features in a very easy and cost efficient way.

Simply connect the Starter Kit board to your PC with the cable supplied, install the MS-Windows® based software and start the pre-programmed demo with a mouse click on one of the installed icons. Due to the extremely low power consumption, no external power supply is needed for this Starter Kit.

But the STK is not only a demo board. You can develop your own program, test it with the basic Simulation Environment and download it in the 512B on-chip RAM of the OTP version to test on board-level. Additionally, it is possible to program the 16KB on-chip OTP memory with your code.

For more information or a complete list of all starter kits available today, please visit http://www.ti.com/sc/msp430

***The MSP430 Starter Kit Package Contains:***

- MSP430 board with OTP, LCD display, PC connector and light sensor
- Cable to connect the STK to your PC
- MS-Windows based basic simulation environment and assembler software*
- Terminal software to communicate with the on-chip monitor
- Starter Kit/Evaluation kit manual
- MSP430 application report
- Architecture users guide and module library
- Software users guide

- Assembly language tools users guide
- MSP430 data sheets
- Simulation environment and LCD editor manual
- MSP430 Brochure

## MSP430 Evaluation Kits

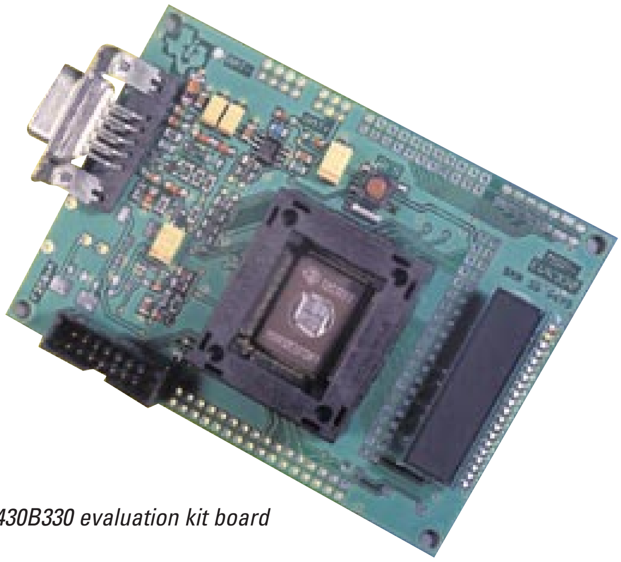
***MSP-EVK430x110, MSP-EVK430x320 and MSP-EVK430x330***

These three evaluation kits are powerful development tools and include much of the hardware and software required to complete your application development. Each EVK (11x, 32x and 33x) comes with its own evaluation board.

The MSP-EVK430x110 evaluation kit supports the MSP430x11x family of devices and comes with the PMS430E112JL (UV EPROM). The MSP-EVK430x320 evaluation kit supports the MSP430x32x family of devices and comes with the PMS430E325FZ (UV EPROM) assembled in a socket. The MSP-EVK430x330 evaluation kit supports the MSP430x33x family of devices and comes with the PMS430E337HFD (UV EPROM).

***The MSP430 Evaluation Kits Contents***

- MSP430 EVK board with UV erasable EPROM, LCD, PC connector



*The MSP-EVK430B330 evaluation kit board*

- Cable to connect the EVK to your PC
- MSP-PRG430 programming adapter*
- Programming adapter software
- MSP-SIM430 simulation environment and assembler language software*
- Terminal software (to communicate with the on-chip monitor)
- Starter-kit/Evaluation kit manual
- Architecture users guide and module library
- Software users guide
- Assembly language tools users guide
- MSP430 Application Report
- MSP430 Data sheets (MSP430x11x, MSP430x31x, MSP430x32x, and MSP430x33x)
- Simulation environment and LCD-editor manual
- MSP430 Brochure

**See the following page for additional information.*

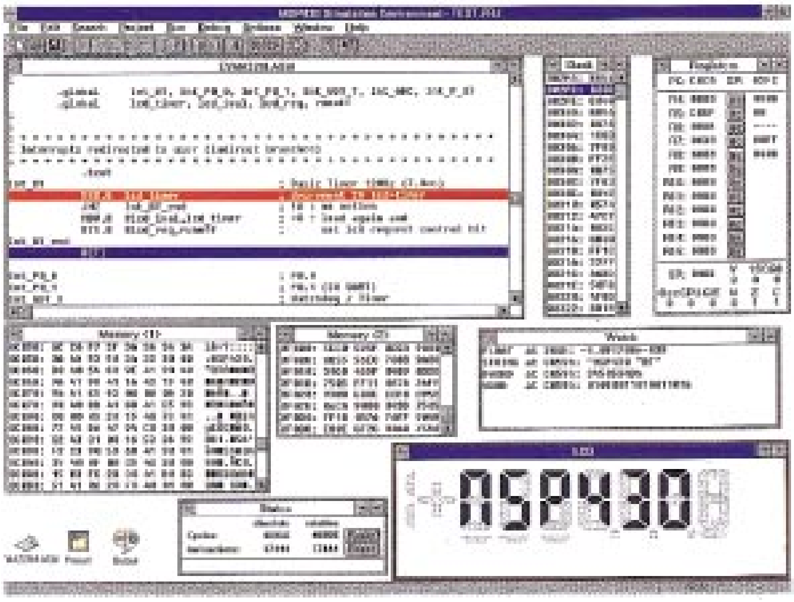
## MSP430 Design Support Features

***Simulation Environment MSP-SIM430***

The MS-Windows based Simulation Environment allows you to get familiar with the MSP430 quickly and easily. The source window allows editing and debugging of your code, along with some breakpoint logic. Several locations in the memory area, e.g. the system stack can be displayed independently.

The Simulation Environment package contains also the MSP430 assembler, which can be used directly from the Simulation Environment.

A custom specific LC display can be simulated with the LCD Designer, which comes in the S/W package.



***MSP-PRG430 Programming Adapter***

To help you program your OTP versions of the MSP430, Texas Instruments offers a Programming Adapter, which can be used for all family members.

It comes together with the programming software including documentation, and allows the in-circuit programming of the devices. The chip is assembled on the final PCB board and can easily be programmed afterwards by connecting the Programming Adapter to the MSP430 JTAG pins; no additional power supply on the board is needed.

***MSP-FPP430 Floating Point Package***

The Floating Point Package for the MSP430 family is supporting the basic arithmetic operations add, subtract, multiply, divide and compare with a 24-bit or 40-bit mantissa. A conversion from/to binary and BCD format is also supported. Support s/w like square roots and trigonometric functions is included as well. The package makes use of the internal registers of the MSP430 RISC CPU; the hardware multiplier can be used in addition.

## MSP430 Third-Party Support

Texas Instruments runs a dedicated MSP430 third-party program for hardware and software tooling as well as for programming support.

These third-party developers have the engineering expertise to provide application-specific software tools to help customers achieve design goals. A hyperlink in the Texas Instruments MSP430 web site will direct you to each company home page. Visit:

**www.ti.com/sc/docs/msp/msp430/430tool.htm**

BP MICROSYSTEMS

BP Microsystems Inc. designs and manufactures device programmers for both engineering and production applications. Leading the industry in device support, performance, and cost of ownership, BP provides complete device programming solutions to customers worldwide. The company offers a full line of single-site device programmers and Universal Programmers, and multi-site Concurrent Programming Systems.

The MSP430 can be programmed using any of our universal engineering programmers (BP-1200 and BP-1400), our manual, multi-site Concurrent Programming Systems.

For more information, please visit the BP web site:

**www.bpmicro.com**

DR. KROHN & STILLER

***Dr. Krohn & Stiller Emulators for Every Budget***

Dr. Krohn & Stiller supports the MSP430 family by a set of three different emulators—one for every budget. All three In-Circuit Emulators (Economy, Standard and Universal) are fully equipped with at least 128K emulation RAM, 32K trace memory and complex trigger capabilities.

A windows debugger interface is available for Windows 3.1, Windows 95 and Windows NT. C code generated by the IAR compiler is directly loaded into emulation memory—no pre-processor is needed. To step through the program the user may choose between C, mixed or assembler display in the HLL debugger. The emulator can even be used as a server for the IAR C-Spy debugger interface. Drag and Drop, coloring of changes in memory or registers, automatically updated local windows, and code coverage, are only a few features of these highly sophisticated emulators.

For more information, please visit the web site of Dr. Krohn & Stiller:

**www.iceworld.de**

### *Göpel electronic On-chip Emulation Software for the MSP430*

This solution is based on a complete software and hardware kit that turns any PC into a powerful device emulator. A standardized IEEE1149.1 / JTAG 4-wire test bus is plugged onto the parallel port via scan controller, while a powerful 32-bit software enables access to the different resources for programming, verification and debug. High efficiency is achieved through interactive control and direct download of the operation software into the on-chip EPROM. With the CPU emulator, debugging of TI assembler code is possible.

A low-cost and true high-level emulator is the result of features such as break-point setting, step function, re-assembler and extended register watching that makes the source code traceable and easy-to-debug. For recording user actions a script language is used. This ASCII language contains commands for memory and register handling, for programming the EPROMS and blowing fuses.

For more information please visit the Göpel web site:

**www.goepel.com**

### *Hitex Emulator in an Affordable Quality Concept*

The MX430 emulator is Hitex's debug tool offered for the MSP430x310 and MSP430x320 configuration. It provides hardware breakpoints in RAM/ROM, simulation memory for internal RAM/ROM with zero wait-states, easy change to other derivates, fast debugging due to transmission rates up to 115200 Baud and more.

The MX430P emulates all derivates of the MSP430x3xx family. The MX430L emulates the MSP430x11x configuration. The new AX430 high-end emulator provides comprehensive debugging and test tools for the whole MSP430x3xx family.

The HiTOP debugger provides comfortable usage guaranteeing high stability. It offers complete symbolic HLL debugging and rapid access to all emulator resources. All commands, such as Line and Single Stepping, GoTo Cursor, GoTo Label and many others can be executed by using the mouse, keyboard or the extensive command language HiSCRIPT.

For more information, please visit the Hitex web site:

**www.hitex.de**

### *IAR C-Cross-Compiler and C/ASM-Debugger*

The EW430 integrates C compiler, linker, librarian and assembler in a seamless environment with easy-to-use project and option handling. The CW430 is a HLL debugger incorporating a complete C expression analyzer and full C-type knowledge. It combines a detailed control of code execution, needed for embedded development debugging, with the flexibility and power of the C language. The source window can display C source code and mix it with assembler.

CW430 simulator allows an unlimited number of breakpoints on C statements, assembler instructions and on any address with an access type of read, write and opcode fetch. Interrupt simulation implements commands to launch specific interrupts at a specific cycle-count or periodically. For interrupt simulation with intermittent interrupts, the same algorithm as the hardware, for choosing the highest priority interrupt to be executed, will be selected. The same interface is available to work on the EVK-Board (CW430R) and on the emulator from Dr. Krohn & Stiller (CW430K), so there is a common interface available for all these tools.

For more information, please visit the IAR web site:

**www.iar.se**

### *SMS Universal Programming System Sprint*

SMS has invested heavily into the steadily growing market of microcontroller devices by considerably expanding its device library. The universal programming systems of the Sprint family do not only stand out from most other programming systems by the variety of supported devices but also by guaranteeing its customers the use of vendor approved algorithms with maximal programming yields and minimal programming times.

SMS supports the TI devices with the universal programmers Sprint Multisystem as well as with the automatic programming systems PP100/22. With the latest version of the Sprint software the following devices are supported: MSP430x313, MSP430x325, MSP430x337 and MSP430x11x. All controllers are supported with flexible TOPs or adapters for different packages like PLCC, SSOP or QFP. Special software functions of the devices, such as UserID, Watchdog Timer or Code Protection can be activated via a simple menu.

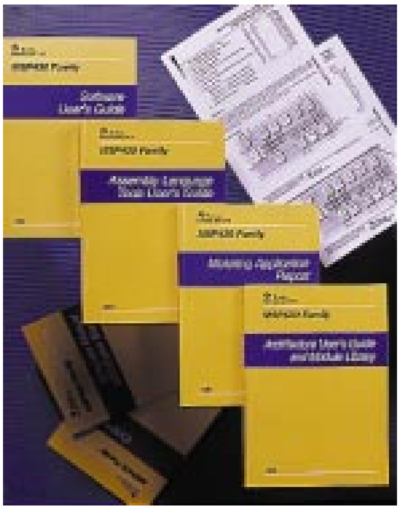
For more information, please visit the SMS web site:

**www.sms-sprint.com**

## Documentation

Texas Instruments provides extensive documentation support for the MSP430 in the form of users guides, applications book, data sheets and brochures.

- Architecture and module library users guide (SLAUE10B)
- Software users guide (SLAUE11)
- Assembly language tools users guide (SLAUE12)
- MSP430 application report (SLAAE10C)
- MSP430x11x data sheet (SLAS196)
- MSP430x31x data sheet (SLAS165B)
- MSP430x32x data sheet (SLAS164 to be replaced by SLAS219 for *new* lower-power versions)
- MSP430x33x data sheet (SLAS163)



The MSP430 application report (SLAAE10C), for example includes many hardware and software examples using the MSP430 for low power, metering, and high precision applications. This report contains examples of developing code and circuitry for most of the hardware found on the MSP430. It will assist you in developing code for ADC, UART, I²C bus, battery check, digital motor control, as well as arithmetic routines, table processing, and much more.

### *For More Information*

For more information about TI's ultra-low-power signal processors, contact us at:

**www.ti.com/sc/msp430**

For a more personal look at the MSP430, or for pricing and availability, contact your local TI sales field office or authorized distributor.

## *TI Worldwide Technical Support*

### Internet

**TI Semiconductor Home Page**
http://www.ti.com/sc

**TI Distributors**
http://www.ti.com/sc/docs/distmenu.htm

### Product Information Centers

**Americas**

| | |
|---|---|
| Phone | +1(972) 644-5580 |
| Fax | +1(972) 480-7800 |
| Email | sc-infomaster@ti.com |

**Europe, Middle East, and Africa**

| | |
|---|---|
| Phone | |
| Deutsch | +49-(0) 8161 80 3311 |
| English | +44-(0) 1604 66 3399 |
| Español | +34-(0) 90 23 54 0 28 |
| Francais | +33-(0) 1-30 70 11 64 |
| Italiano | +33-(0) 1-30 70 11 67 |
| Fax | +44-(0) 1604 66 33 34 |
| Email | epic@ti.com |

**Japan**

| | |
|---|---|
| Phone | |
| International | +81-3-3457-0972 |
| Domestic | 0120-81-0026 |
| Fax | |
| International | +81-3-3457-1259 |
| Domestic | 0120-81-0036 |
| Email | pic-japan@ti.com |

**Asia**

| | | |
|---|---|---|
| Phone | | |
| International | +886-2-23786800 | |
| Domestic | Local Access Code | TI Number |
| Australia | 1-800-881-011 | -800-800-1450 |
| China | 10810 | -800-800-1450 |
| Hong Kong | 800-96-1111 | -800-800-1450 |
| India | 000-117 | -800-800-1450 |
| Indonesia | 001-801-10 | -800-800-1450 |
| Korea | 080-551-2804 | - |
| Malaysia | 1-800-800-011 | -800-800-1450 |
| New Zealand | 000-911 | -800-800-1450 |
| Philippines | 105-11 | -800-800-1450 |
| Singapore | 800-0111-111 | -800-800-1450 |
| Taiwan | 080-006800 | - |
| Thailand | 0019-991-1111 | -800-800-1450 |
| Fax | 886-2-2378-6808 | |
| Email | tiasia@ti.com | |



B010899

Windows is a registered trademark of Microsoft Corporation.

TEXAS INSTRUMENTS



Printed in the U.S.A. at ColorDynamics, Allen, TX

# LoRa® and LoRaWAN®

# Table of Contents

# List of figures

AN1200.86

# List of tables

# Chapter 1. What are LoRa® and LoRaWAN®?

LoRa is an RF modulation technology for low-power, wide area networks (LPWANs). The name, LoRa, is a reference to the extremely long-range data links that this technology enables. Created by Semtech to standardize LPWANs, LoRa provides for long-range communications: up to three miles (five kilometers) in urban areas, and up to 10 miles (15 kilometers) or more in rural areas (line of sight). A key characteristic of the LoRa-based solutions is ultra-low power requirements, which allows for the creation of battery-operated devices that can last for up to 10 years. Deployed in a star topology, a network based on the open LoRaWAN protocol is perfect for applications that require long-range or deep in-building communication among a large number of devices that have low power requirements and that collect small amounts of data.

Consider the differences between LoRa and other network technologies that are typically used in IoT or traditional machine-to-machine (M2M) connectivity solutions:

| | | |
|---|---|---|
| Traditional Cellular<br><br>Long Range<br>High Data Rates<br>Low Battery Life<br>High Cost | LoRa®<br>LPWAN (3-5B in 2022)<br><br>Long Range<br>Low Data Rates<br>Long Battery Life<br>Low Cost<br>High Capacity Potential | Cat-M1<br><br>Long Range<br>High Data Rates<br>Low Battery Life<br>Medium Cost |
| Local Area Network<br>(Wi-Fi)<br><br>Short Range<br>High Data Rates<br>Low Battery Life<br>Medium Cost | Narrow-Band IoT<br>(NB-IoT)<br><br>Stationary Devices<br>Short Range (indoor coverage)<br>Low Data Rates<br>Good Battery Life<br>Low Cost | Personal Area Network<br>(Bluetooth®)<br><br>Very Short Range<br>Low data rates<br>Good Battery Life<br>Low Cost |

*Figure 1. IoT Technologies*

| | |
|---|---|
| **NOTE** | In Europe, mobile network operators have implemented a dual strategy to address packet size and latency issues. They often offer both LoRaWAN and Cat-M1, which are complementary technologies. LoRaWAN accommodates the need for longer battery life, with a trade-off of longer latency and smaller packet sizes. In contrast Cat-M1 can be used for larger payloads with less latency than LoRaWAN can accommodate. |

Figure 2 highlights some important advantages of deploying a LoRaWAN network:

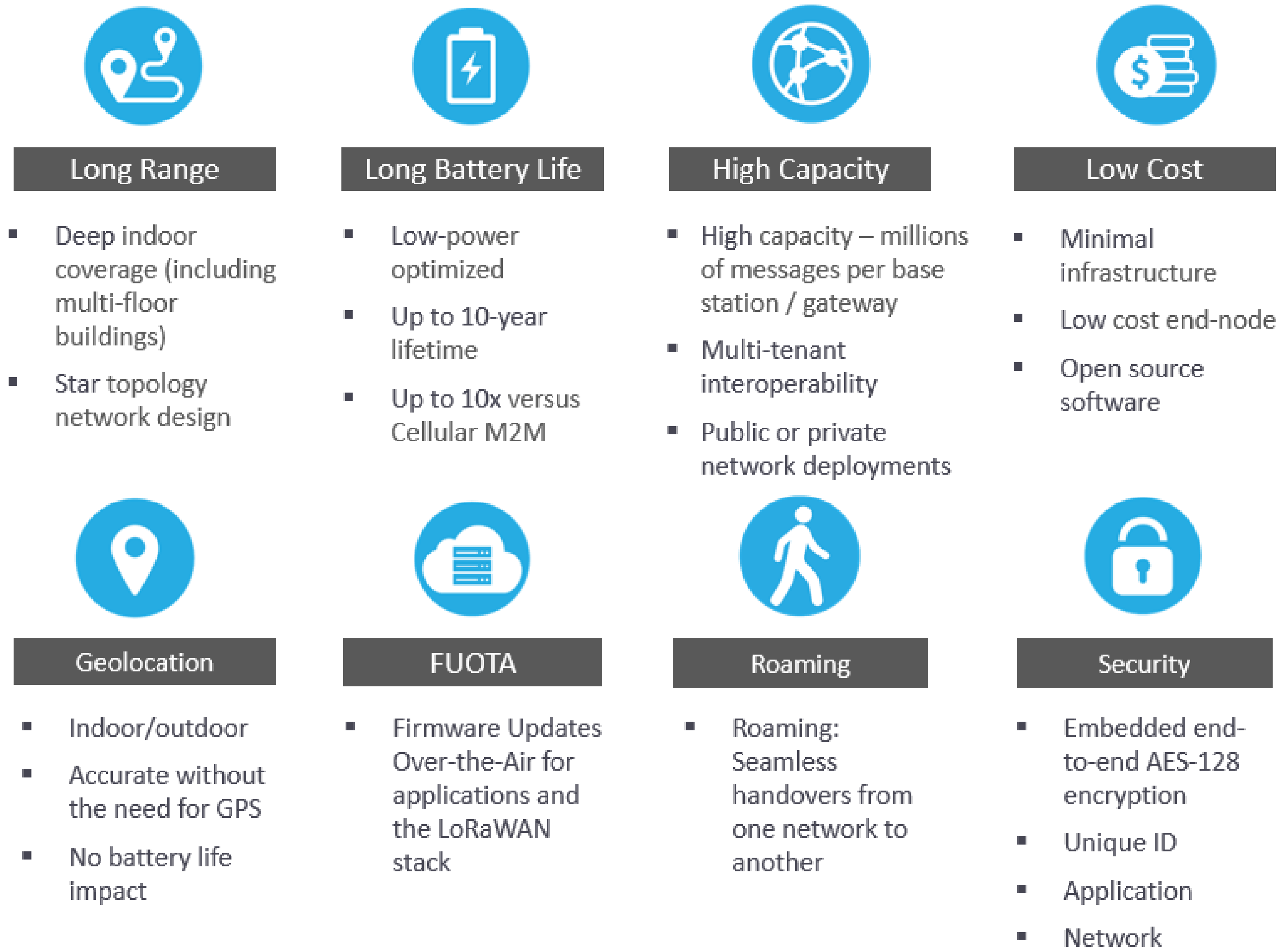



*Figure 2. Advantages of deploying a LoRaWAN network*

Let's look into these advantages in a little more depth.

With respect to range, a single LoRa-based gateway can receive and transmit signals over a distance of more than 10 miles (15 kilometers) in rural areas. Even in dense urban environments, messages are able to travel up to three miles (five kilometers), depending on how deep indoors the end devices (end nodes) are located.

As far as battery life goes, the energy required to transmit a data packet is quite minimal given that the data packets are very small and only transmitted a few times a day. Furthermore, when the end devices are asleep, the power consumption is measured in milliwatts (mW), allowing a device's battery to last for many, many years.

When it comes to capacity, a LoRaWAN network can support millions of messages. However, the number of messages supported in any given deployment depends upon the number of gateways that are installed. A single eight-channel gateway can support a few hundred thousand messages over the course of a 24-hour period. If each end device sends 10 messages a day, such a gateway can support about 10,000 devices.[1] If the network includes 10 such gateways, the network can support roughly 100,000 devices and one million messages. If more capacity is required, all that is needed is to add additional gateways to the network.

And then, there is cost. Given the capabilities of LoRa-based end nodes and gateways, only a few gateways - configured in a star network - are required

AN1200.86

to serve a multitude of end nodes. This means that capital and operational expenses can be kept relatively low. Also, when the cost-effective LoRa RF modules that are embedded in inexpensive end nodes are used in conjunction with the open LoRaWAN standard, the return on investment can be considerable.

[1] There is no one-to-one relationship between LoRa-based devices and gateways in a LoRaWAN network; messages sent to and from end devices travel through all gateways within range. De-duplication is handled by the network server.

# Chapter 2. Radio Modulation and LoRa

A proprietary spread-spectrum modulation technique derived from existing Chirp Spread Spectrum (CSS) technology, LoRa offers a trade-off between sensitivity and data rate, while operating in a fixed-bandwidth channel of either 125 KHz or 500 KHz (for uplink channels), and 500 KHz (for downlink channels). Additionally, LoRa uses orthogonal spreading factors. This allows the network to preserve the battery life of connected end nodes by making adaptive optimizations of an individual end node's power levels and data rates. For example, an end device located close to a gateway should transmit data at a low spreading factor, since very little link budget is needed. However, an end device located several miles from a gateway will need to transmit with a much higher spreading factor. This higher spreading factor provides increased processing gain, and higher reception sensitivity, although the data rate will, necessarily, be lower.

LoRa is purely a physical (PHY), or "bits" layer implementation, as defined by the OSI seven-layer Network Model, depicted in Figure 3. Instead of cabling, the air is used as a medium for transporting LoRa radio waves from an RF transmitter in an IoT device to an RF receiver in a gateway, and vice versa.

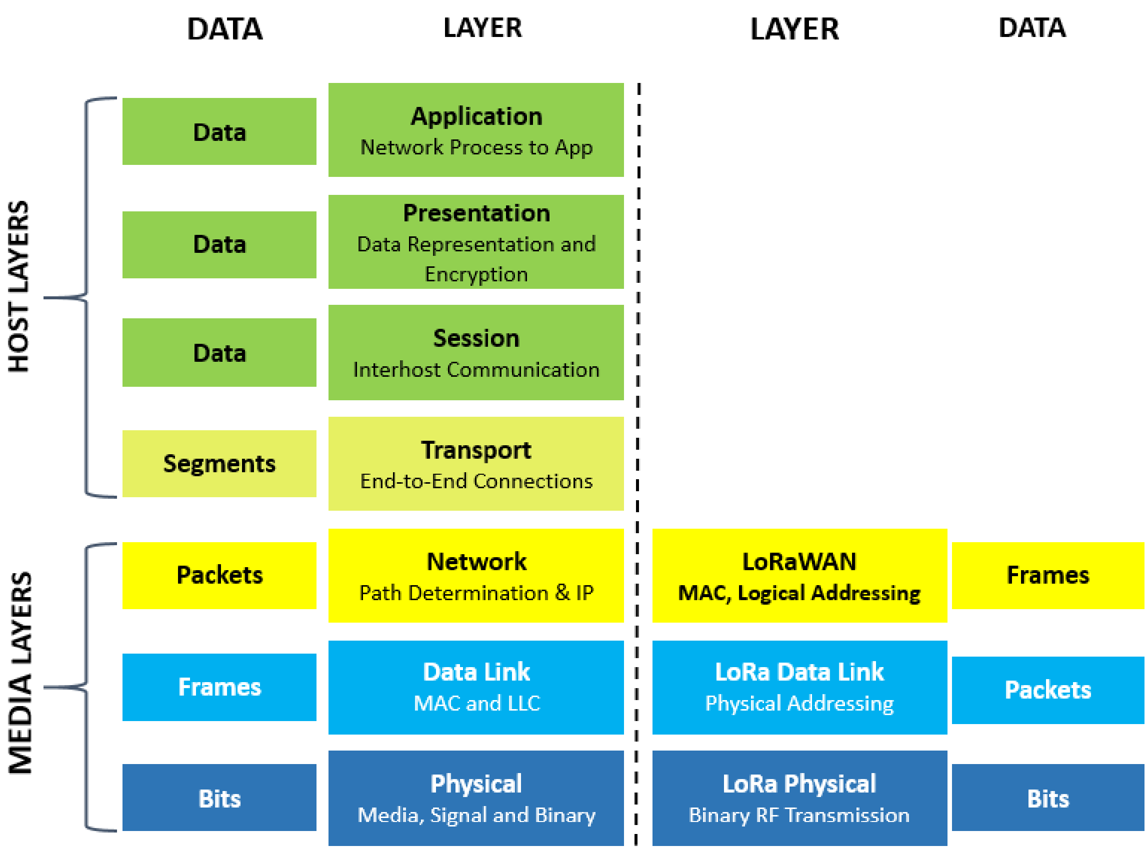



*Figure 3. OSI seven-layer network model*

In a traditional or Direct Sequence Spread Spectrum (DSSS) system, the carrier phase of the transmitter signal changes according to a code sequence as shown in Figure 4. When multiplying the data signal with a pre-defined bit pattern at a much higher rate, also known as a spreading code (or chip sequence), a "faster" signal is created that has higher frequency components than the original data signal. This means that the signal bandwidth is spread beyond the bandwidth of the original signal. In RF terminology, the bits of the code sequence are called chips (in order to distinguish between the longer, un-coded, bits of the original data signal). When the transmitted signal arrives at the RF receiver, it is multiplied with an identical copy of the spreading code used in the RF transmitter, resulting in a replica of the original data signal.

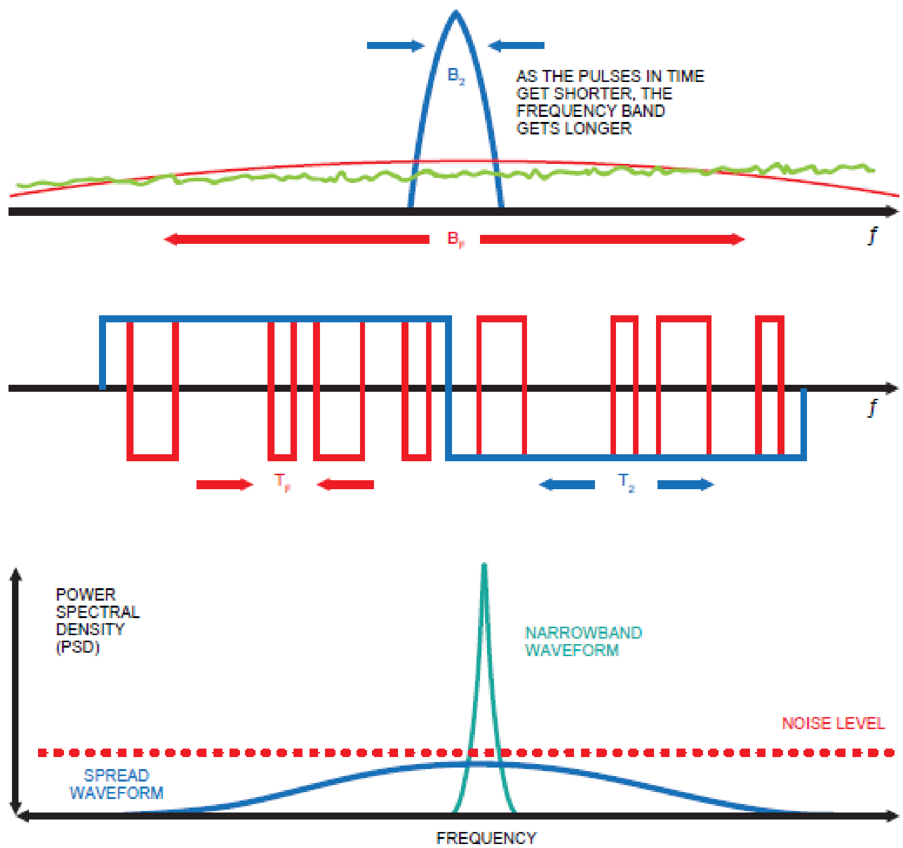



*Figure 4. DSSS system carrier phase transmitter signal changes*

You might ask: Why go through all this trouble? Why not just transmit the original data signal instead of going through this code sequence multiplication? The answer is simple: going through this code sequence multiplication buys you a higher RF link budget, so you can transmit over a longer range.

The Log10 ratio of the code sequence's chip rate and the data signal's bit rate is called the processing gain (Gp). This gain is what allows the receiver to recover the original data signal, even if the channel has a negative signal-to-noise ratio (SNR). LoRa has a superior Gp compared to frequency-shift keying (FSK) modulation, allowing for a reduced transmitter output power level while maintaining the same signal data rate and a similar link budget.

One of the downsides of a DSSS system is the fact that it requires a highly-accurate (and expensive) reference clock. Semtech's LoRa Chirp Spread Spectrum (CSS) technology offers a low-cost and low-power, yet robust, DSSS alternative that does not require a highly-accurate reference clock. In LoRa modulation, the spreading of the signal's spectrum is achieved by generating a chirp signal that continuously varies in frequency, as is depicted in Figure 5.

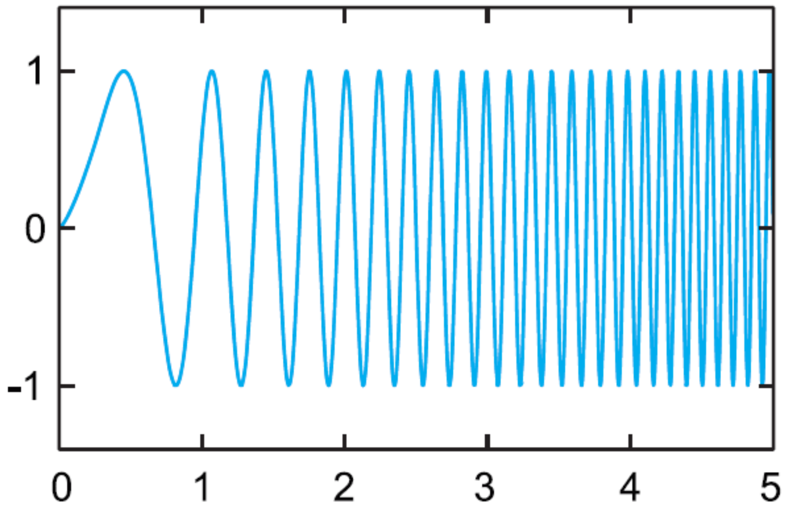



*Figure 5. LoRa Chirp Spread Spectrum illustration*

An advantage of this method is that the timing and frequency offsets between transmitter and receiver are equivalent, greatly reducing the complexity of the receiver design. The frequency bandwidth of this chirp is equivalent to the spectral bandwidth of the signal. The data signal that carries the data from an end device to a gateway is chipped at a higher data rate and modulated onto the chirp carrier signal. LoRa modulation also includes a variable error correction scheme that improves the robustness of the transmitted signal. For every four bits of information sent, a fifth bit of parity information is sent.

## 2.1. Key LoRa Modulation Properties

As noted above, LoRa processing gain is introduced in the RF channel by multiplying the data signal with a spreading code or chip sequence. By increasing the chip rate, we increase the frequency components of the total signal spectrum. In other words, the energy of the total signal is now spread over a wider range of frequencies, allowing the receiver to discern a signal with a lower (that is, worse) signal-to-noise ratio (SNR).

In LoRa terms, the amount of spreading code applied to the original data signal is called the *spreading factor* (SF). LoRa modulation has a total of six spreading factors (SF7 to SF12). The larger the spreading factor used, the farther the signal will be able to travel and still be received without errors by the RF receiver.

Figure 6 shows the four different spreading factors [SF7...SF10] that can be used for uplink (UL) messages on a 125 KHz channel.[1] It shows the equivalent bit rate as well as the estimated range (this depends on the terrain; longer distances will be achieved in a rural environment than in an urban environment). It also shows the dwell time, or *time on air* (TOA), values for an 11-byte payload for each of the four spreading factors.

| Spreading Factor (For UL at 125 KHz) | Bit Rate | Range (Depends on Terrain) | Time on Air for an 11-byte payload |
|---|---|---|---|
| SF10 | 980 bps | 8 km | 371 ms |
| SF9 | 1760 bps | 6 km | 185 ms |
| SF8 | 3125 bps | 4 km | 103 ms |
| SF7 | 5470 bps | 2 km | 61 ms |

*Figure 6. LoRa Spreading Factors*

Importantly, the LoRa modulation spreading factors are inherently orthogonal. This means that signals modulated with different spreading factors and transmitted on the same frequency channel at the same time do not interfere with each other. Instead, signals at different spreading factors simply appear to be noise to each other.

LoRa signals are robust and very resistant to both in-band and out-of-band interference mechanisms. LoRa modulation also offers immunity to multipath and fading, making it ideal for use in urban and suburban environments, where both mechanisms dominate. Additionally, Doppler shifts cause a small frequency shift in the time axis of the baseband signal. This frequency offset tolerance mitigates the requirement for tight tolerance reference clock sources and, therefore, makes LoRa ideal for data communications from devices that are mobile.

## 2.2. LoRa Modulation Characteristics

The LoRa modulation characteristics for each region are defined in the LoRaWAN Regional Parameters document, available from the LoRa Alliance®. In North America, there are 64, 125 kHz LoRa uplink channels defined, centered on a 200 kHz raster as can be seen in Figure 7. There are eight 500 kHz uplink channels as well as eight, 500 kHz downlink channels defined. In North America, gateways can have up to 64, 125 kHz uplink channels as well as eight 500 kHz uplink and downlink channels. This type of gateway is referred to as a carrier grade macro gateway and is used for outdoor applications only.

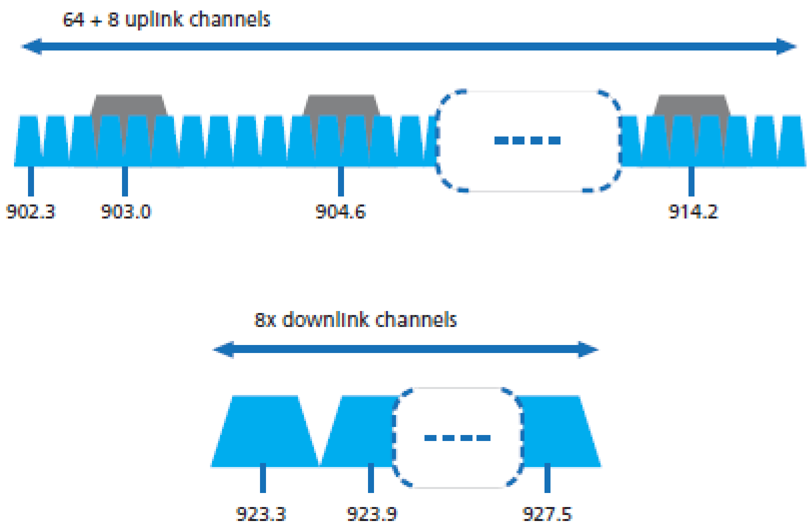



*Figure 7. LoRaWAN North America Channel Plan*

Figure 8 provides another way to understand these modulation characteristics.

| Data Rate (DR) | Spreading Factor (SF) | Channel Frequency | Uplink or Downlink | Bitrate (Bits/Sec) | Maximum User Payload Size (Bytes) |
|---|---|---|---|---|---|
| 0 | SF10 | 125 kHz | Uplink | 980 | 11 |
| 1 | SF9 | 125 kHz | Uplink | 1,760 | 53 |
| 2 | SF8 | 125 kHz | Uplink | 3,125 | 125 |
| 3 | SF7 | 125 kHz | Uplink | 5,470 | 242 |
| 4 | SF8 | 500 kHz | Uplink | 12,500 | 242 |
| 5 – 7 | | | | | |
| 8 | SF12 | 500 kHz | Downlink | 980 | 53 |
| 9 | SF11 | 500 kHz | Downlink | 1,760 | 129 |
| 10 | SF10 | 500 kHz | Downlink | 3,125 | 242 |
| 11 | SF9 | 500 kHz | Downlink | 5,470 | 242 |
| 12 | SF8 | 500 kHz | Downlink | 12,500 | 242 |
| 13 | SF8 | 500 kHz | Downlink | 21,900 | 242 |

*Figure 8. LoRa modulation characteristics*

- The LoRa physical layer is intended for low throughput, low data rate, and high link budget (i.e., “long-range”) applications.
- For a fixed channel bandwidth, the higher the spreading factor, the higher the processing gain, resulting in an increase in sensitivity and, therefore, an increase in link budget. Subsequently, however, the time on air will also increase.
- Orthogonality between spreading factors allows for the transmission of multiple LoRa signals that are both on the same channel frequency **and** in the same time-slot.
- For a fixed SF, a narrower bandwidth will increase sensitivity as the bit rate is reduced.
- LoRaWAN in North America uses 125 kHz uplink channels and 500 kHz uplink and downlink channels
- The Code Rate is the degree of redundancy implemented by the forward error correction (FEC) used to detect errors and correct them. This rate is fixed at 4/5 for the LoRaWAN protocol

As Stephan Hengstler asserts in his book, *A Novel Chirp Modulation Spread Spectrum technique for Multiple Access*, “LoRa is a constant envelope modulation (very low cost, power efficient power amplifier implementation) … [it] is the most robust, ultra-low power and long range RF solution available.”

## 2.3. Data Collisions and Spreading Factor Orthogonality

With LoRa, packets using different spreading factors are orthogonal, meaning that they are invisible to each other: as mentioned earlier, they simply appear as noise to one another. Therefore, two packets that arrive at the same time on the same receive channel at different spreading factors will not collide and, both will be demodulated by the gateway modem chip. However, two packets with the *same* spreading factor arriving at the same time on the same channel **might** result in a collision. However if one of the two packets is stronger by six dB, it will survive.

The capacity of a LoRaWAN network is a function of its gateway density. To maximize the capacity of the network, using an adaptive data rate (ADR) mechanism is essential. The main goal of ADR is to save the battery power of the LoRaWAN end-nodes. By having the end-nodes closest to a gateway transmit using the lowest spreading factor, their time on air is minimized, thereby prolonging their battery life. More distant sensors transmit at a higher

spreading factor. A trade-off is made between battery power and distance given that a higher spreading factor allows for a gateway to connect to devices that are farther away.

[1] Downlink messages broadcast over 500 KHz channels can use all six available spreading factors (SF7…SF12).

# Chapter 3. LoRaWAN Network Fundamentals

To fully understand LoRaWAN networks, we will start with a look at the technology stack. As shown in Figure 9, LoRa is the physical (PHY) layer, i.e., the wireless modulation used to create the long-range communication link. LoRaWAN is an open networking protocol that delivers secure bi-directional communication, mobility, and localization services standardized and maintained by the LoRa Alliance.

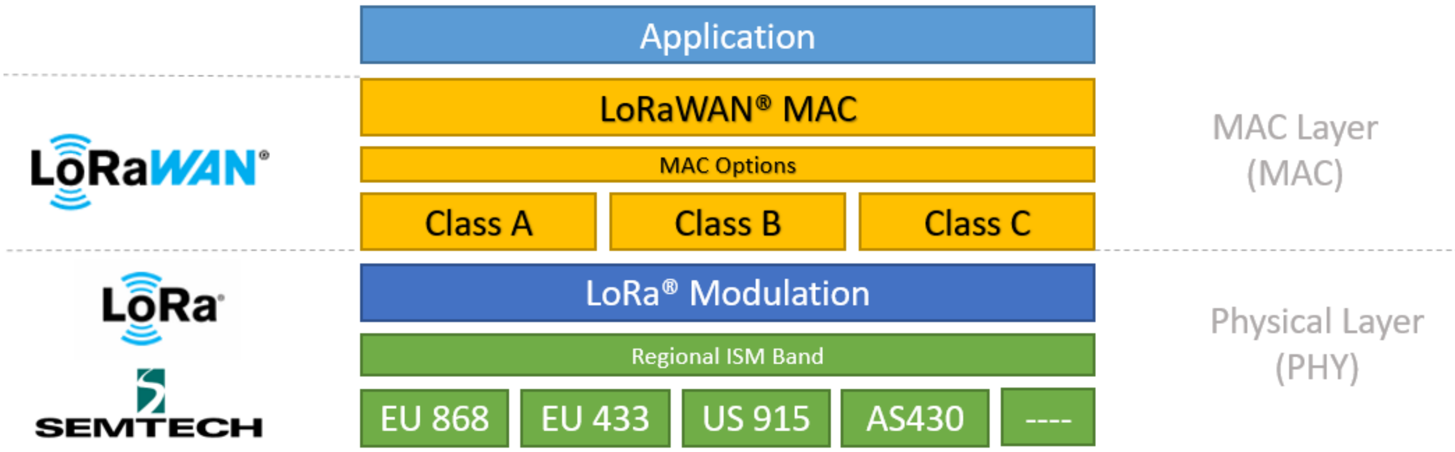



*Figure 9. LoRaWAN technology stack*

## 3.1. LoRaWAN Network Elements: An Introduction

Now that we have a basic understanding of LoRa, we will examine the architecture of a LoRaWAN network. Figure 10 shows a typical LoRaWAN network implementation from end to end.

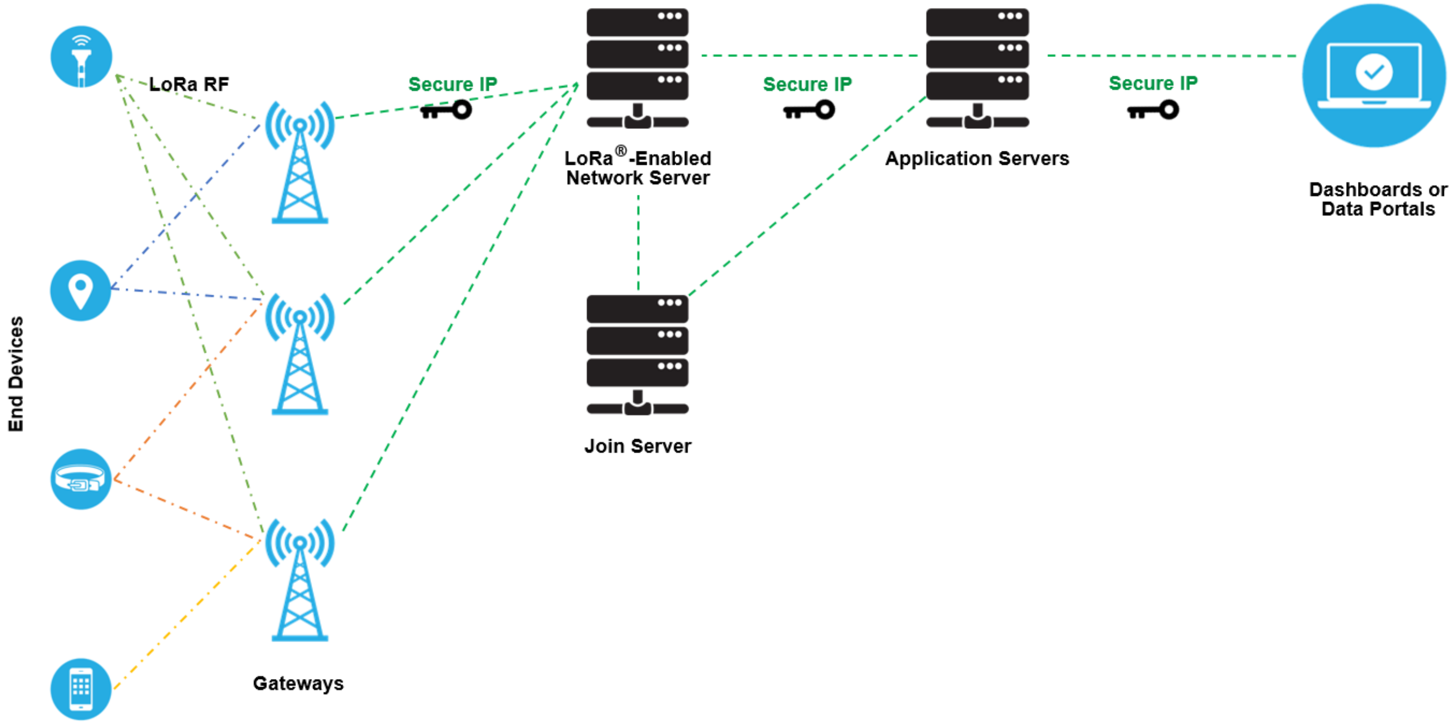



*Figure 10. Typical LoRaWAN network implementation*

Let us examine this diagram in smaller pieces.

## 3.1.1. LoRa-based End Devices

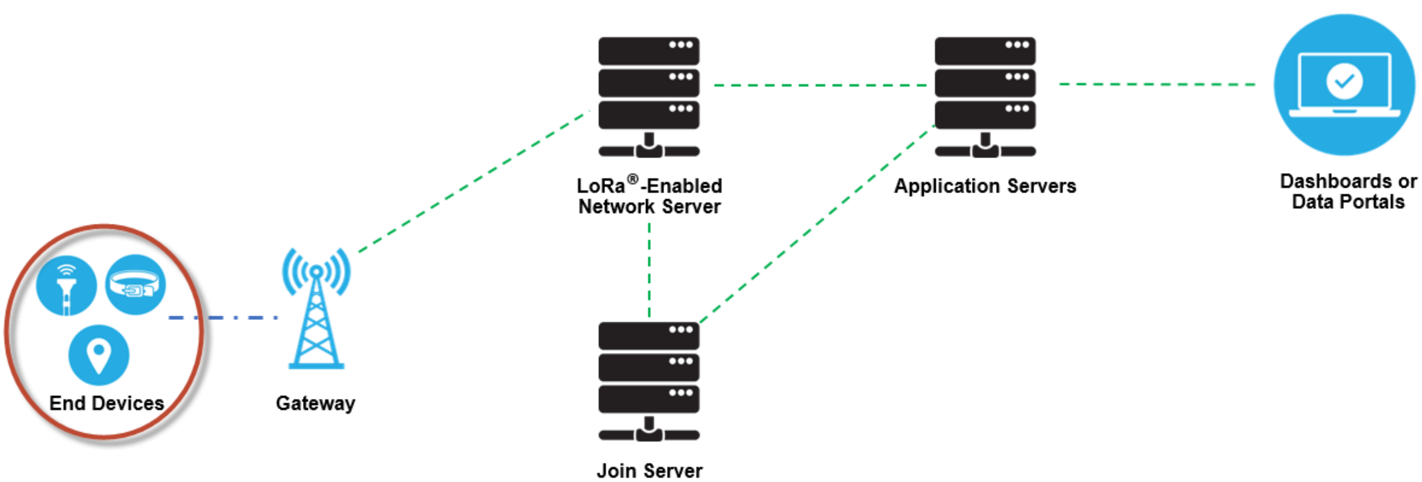



*Figure 11. End devices in a typical LoRaWAN network deployment*

A LoRaWAN-enabled **end device** is a sensor or an actuator which is wirelessly connected to a LoRaWAN network through radio gateways using LoRa RF Modulation.

In the majority of applications, an end device is an autonomous, often battery-operated sensor that digitizes physical conditions and environmental events. Typical use cases for an actuator include: street lighting, wireless locks, water valve shut off, leak prevention, among others.

When they are being manufactured, LoRa-based devices are assigned several unique identifiers. These identifiers are used to securely activate and administer the device, to ensure the safe transport of packets over a private or public network and to deliver encrypted data to the Cloud.

### 3.1.2. LoRaWAN Gateways

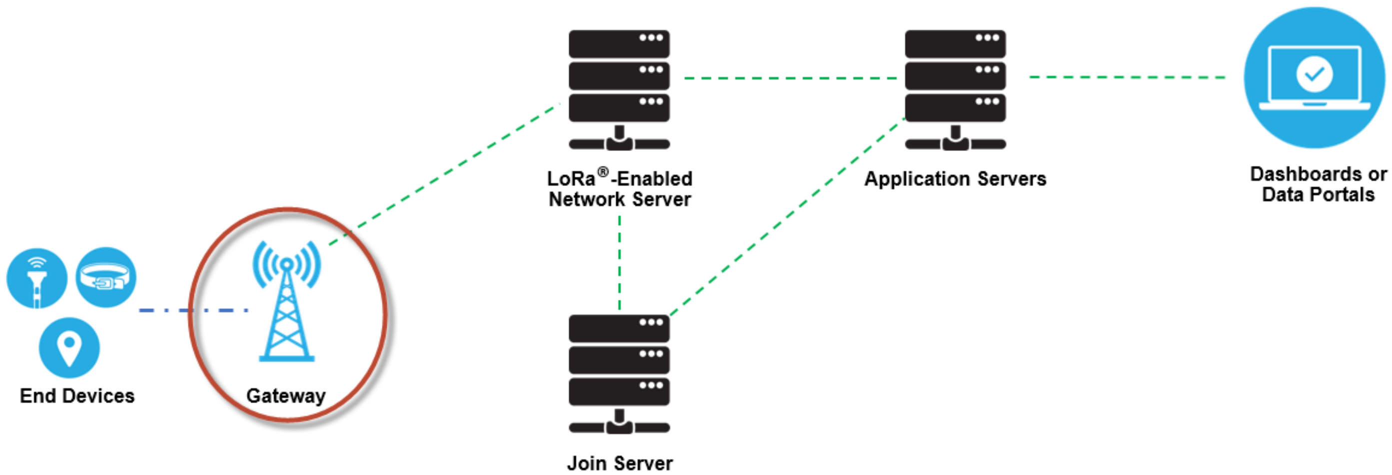



*Figure 12. Gateways in a typical LoRaWAN network deployment*

A LoRaWAN **gateway** receives LoRa modulated RF messages from any end device in hearing distance and forwards these data messages to the LoRaWAN network server (LNS), which is connected through an IP backbone. There is no fixed association between an end device and a specific gateway. Instead, the same sensor can be served by multiple gateways in the area. With LoRaWAN, each uplink packet sent by the end-device will be received by all gateways within reach, as illustrated in Figure 12. This arrangement significantly reduces packet error rate (since the chances that at least one gateway will receive the message are very high), significantly reduces battery overhead for mobile/nomadic sensors, and allows for low-cost geolocation (assuming the gateways in question are geolocation-capable).

The IP traffic from a gateway to the network server can be backhauled via Wi-Fi, hardwired Ethernet or via a Cellular connection. LoRaWAN gateways operate entirely at the physical layer and, in essence, are nothing but LoRa radio message forwarders. They only check the data integrity of each incoming LoRa RF message. If the integrity is not intact, that is, if the CRC is incorrect, the message will be dropped. If correct the gateway will forward it to the LNS, together with some metadata that includes the receive RSSI level of the message as well as an optional timestamp. For LoRaWAN downlinks, a gateway executes transmission requests coming from the LNS without any interpretation of the payload. Since multiple gateways can receive the same LoRa RF message from a single end device, the LNS performs data de-duplication and deletes all copies. Based on the RSSI levels of the identical messages, the network server typically selects the gateway that received the message with the best RSSI when transmitting a downlink message because that gateway is the one closest to the end device in question.

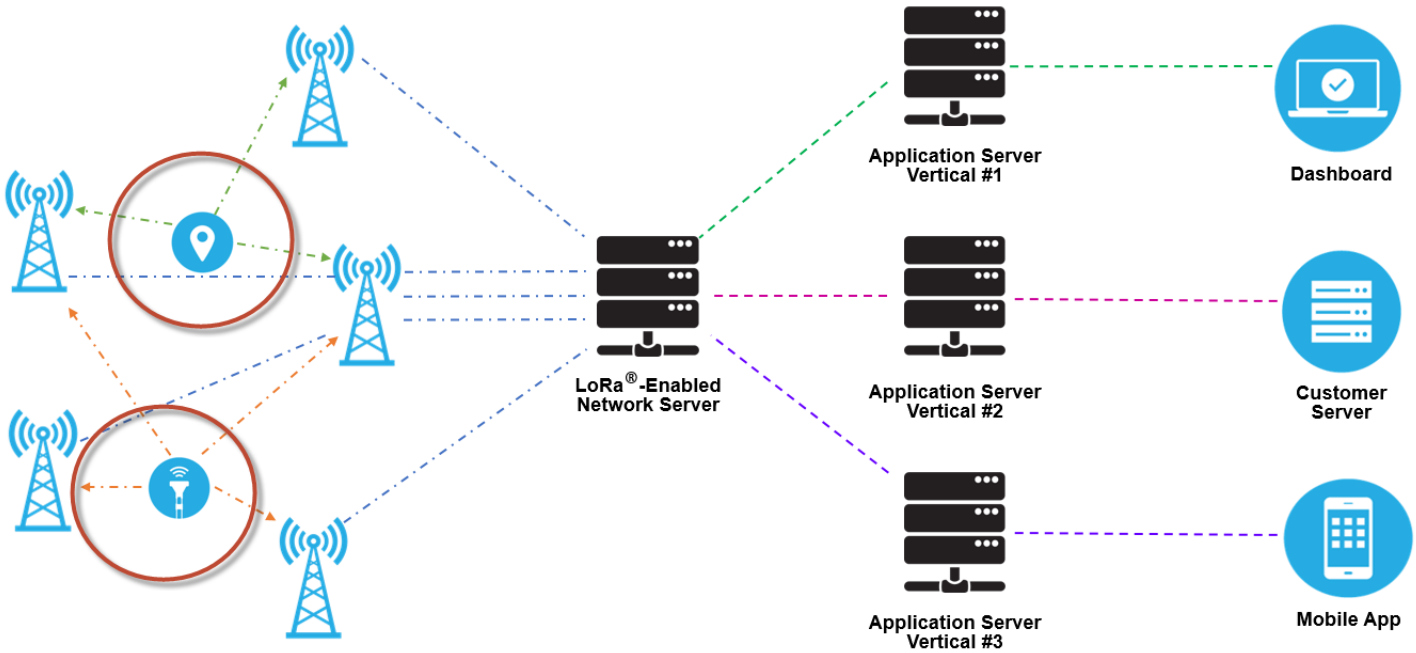



*Figure 13. Gateways receiving and transmitting messages from end devices*

Furthermore, LoRa allows for scalable, cost-optimized gateway implementation, depending on deployment objectives. For example, in North America, 8-, 16-, and 64-channel gateways are available.

The 8-channel gateways are the least expensive. The type of gateway needed will depend on the use case. Eight- and 16-channel gateways are available for both indoor and outdoor use. Sixty-four channel gateways are only available in a carrier-grade variant. This type of gateway is intended for deployment in such places as cell towers, the rooftops of very tall buildings, etc.

## 3.1.3. Network Server

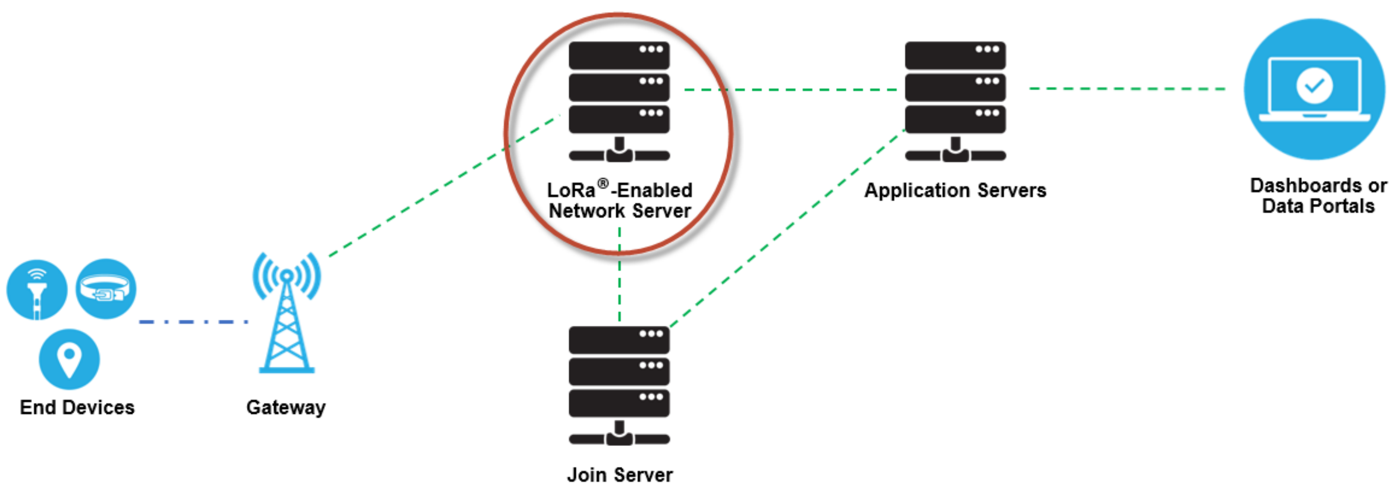



*Figure 14. LoRaWAN Network Server in a typical LoRaWAN network deployment*

The LoRaWAN network server (LNS) manages the entire network, dynamically controls the network parameters to adapt the system to ever-changing conditions, and establishes secure 128-bit AES connections for the transport of both the end to end data (from LoRaWAN end device to the end users

Application in the Cloud) as well as for the control of traffic that flows from the LoRaWAN end device to the LNS (and back). The network server ensures the authenticity of every sensor on the network and the integrity of every message. At the same time, the network server cannot see or access the application data.

In general, all LoRaWAN network servers share the following features:

- Device address checking
- Frame authentication and frame counter management
- Acknowledgements of received messages
- Adapting data rates using the ADR protocol
- Responding to all MAC layer requests coming from the device,
- Forwarding uplink application payloads to the appropriate application servers
- Queuing of downlink payloads coming from any Application Server to any device connected to the network
- Forwarding Join-request and Join-accept messages between the devices and the join server

## 3.1.4. Application Servers

Application servers are responsible for securely handling, managing and interpreting sensor application data. They also generate all the application-layer downlink payloads to the connected end devices.

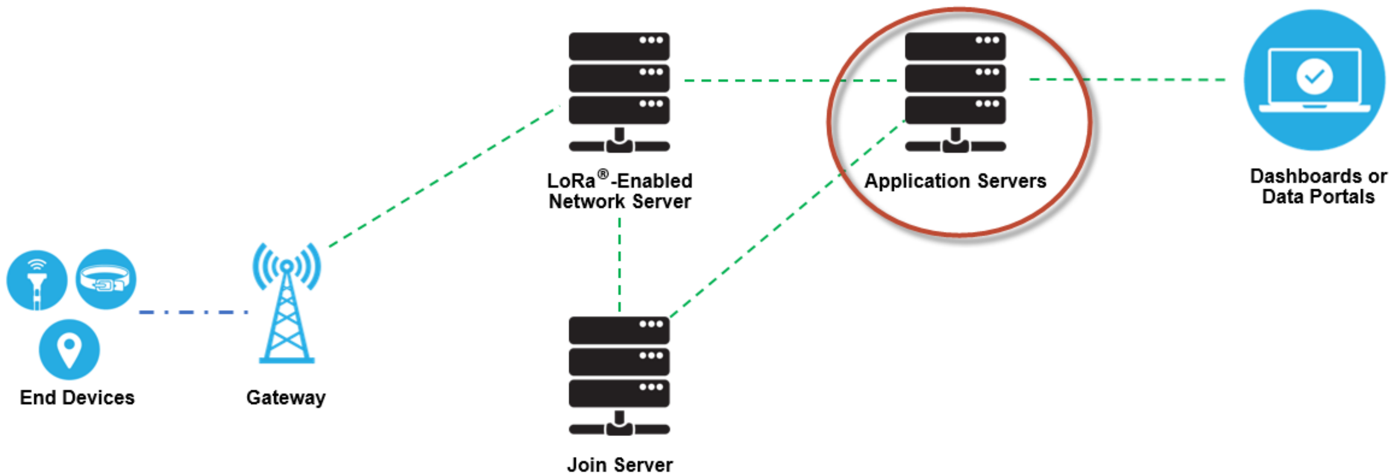



*Figure 15. LoRaWAN Application Server in a typical LoRaWAN network deployment*

## 3.1.5. Join Server

The join server manages the over-the-air activation process for end devices to be added to the network.

The join server contains the information required to process uplink *join-request* frames and generate the downlink *join-accept* frames. It signals to the network server which application server should be connected to the end-device, and performs the network and application session encryption key derivations. It communicates the Network Session Key of the device to the network server, and the Application Session Key to the corresponding application server

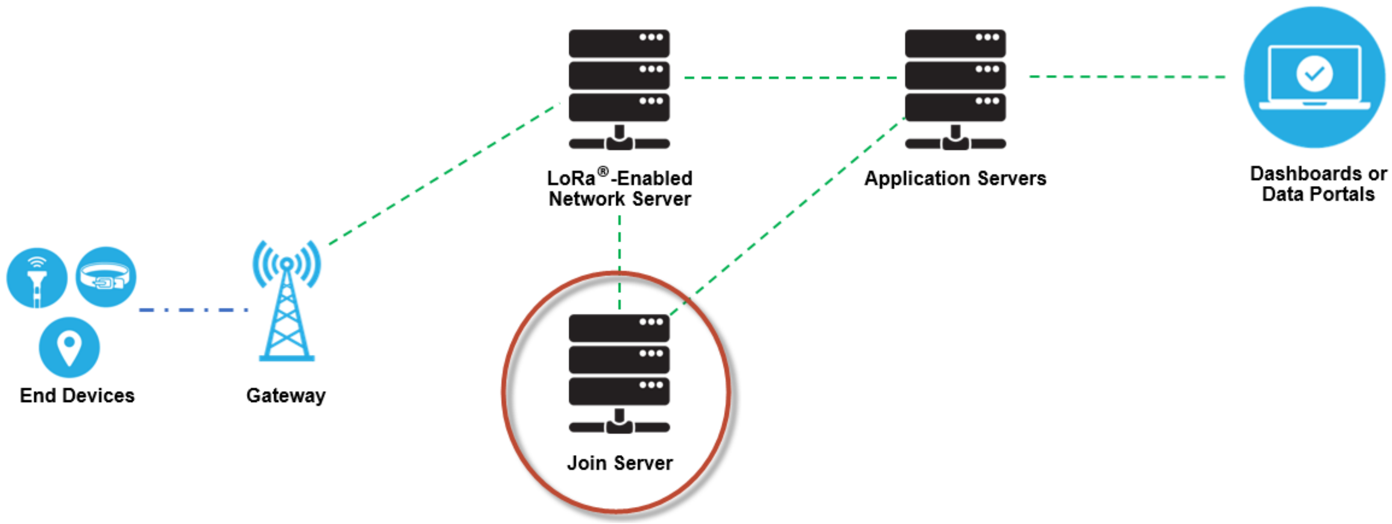



*Figure 16. LoRaWAN Join Server in a typical LoRaWAN network deployment*

For that purpose, the join server must contain the following information for each end-device under its control:

- DevEUI (end-device serial unique identifier)
- AppKey (application encryption key)
- NwkKey (network encryption key)
- Application Server identifier
- End-Device Service Profile

## 3.2. LoRaWAN Network Elements: Device Commissioning

For the sake of security, quality of service, billing, and other needs, devices must be commissioned and activated on the network at the start of operation. The commissioning process securely aligns each device and the network with respect to essential provisioning parameters (such as identifiers, encryption keys, and server locations)

The LoRaWAN specification allows for two types of activation: Over-the-Air Activation (OTAA) (preferred) and Activation by Personalization (ABP). Figure 17 shows the different characteristics of each of these types of activation.

| Over-the-Air Activation (OTAA) | Activation by Personalization (ABP) |
|---|---|
| • Device manufacturers autonomously generate essential provisioning parameters<br>• Secure keys (session-long and derived) can be renewed regularly<br>• Devices can store multiple "identities" to dynamically and securely switch networks and operators during its lifetime<br>• High-grade, tamper-proof security options are available | • A simplified (less secure) commissioning process<br>• IDs and Keys are personalized at fabrication<br>• Devices become immediately functional upon powering up; the Join procedure is skipped<br>• Devices are tied to a specific network/service; the NetID is a portion of the device network address |

*Figure 17. Activation Types*

## 3.3. LoRaWAN Network Elements: Security

There are two key elements to the security of a LoRaWAN network: the *join procedure* and *message authentication*. The join procedure establishes mutual authentication between an end device and the LoRaWAN network to which it is connected. Only authorized devices are allowed to join the network. LoRaWAN MAC and application messages are origin-authenticated, integrity-protected and encrypted end-to-end (i.e., from end device to the application server and vice versa).

These security features ensure that:

- Network traffic has not been altered
- Only legitimate devices are connected to the LoRaWAN network
- Network traffic cannot be listened to (no eavesdropping)
- Network traffic cannot be captured and replayed

With that foundation, we will take a look at the LoRaWAN security measures in more detail.

### 3.3.1. The Join Procedure

We will begin with the security keys, as illustrated in Figure 18. Individual root keys are securely stored on the end devices, and matching keys are securely stored on the join server.

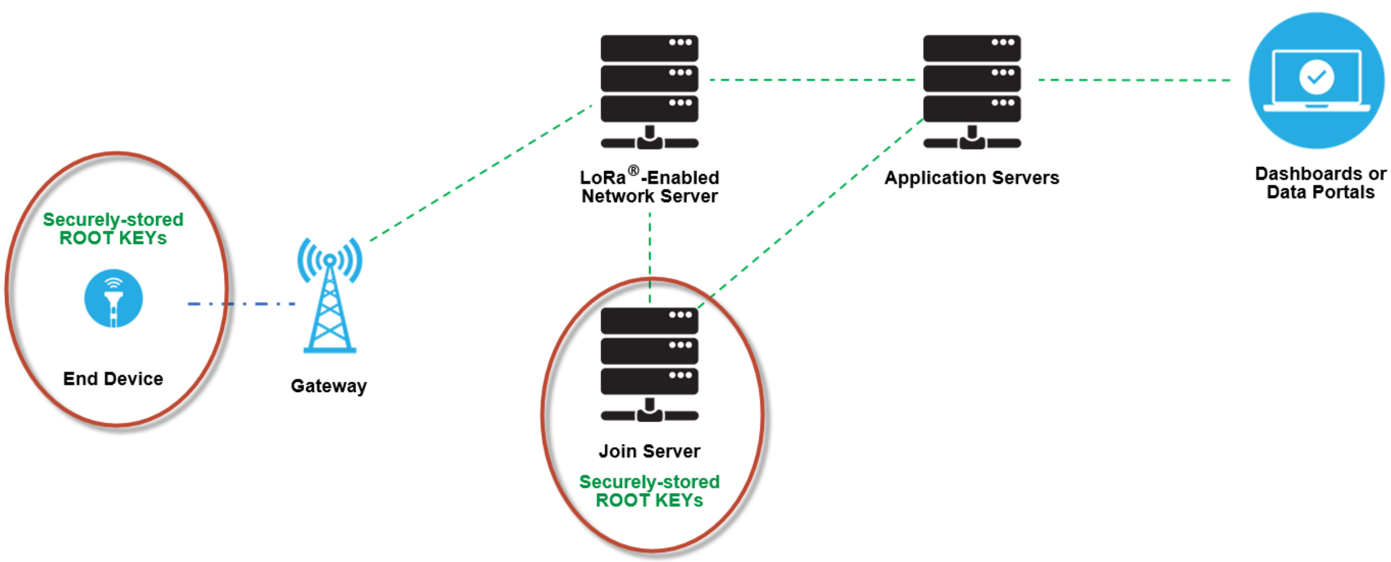



*Figure 18. Security keys generated during the Join procedure*

The end device sends a *join request* message to the join server, as illustrated in Figure 19.

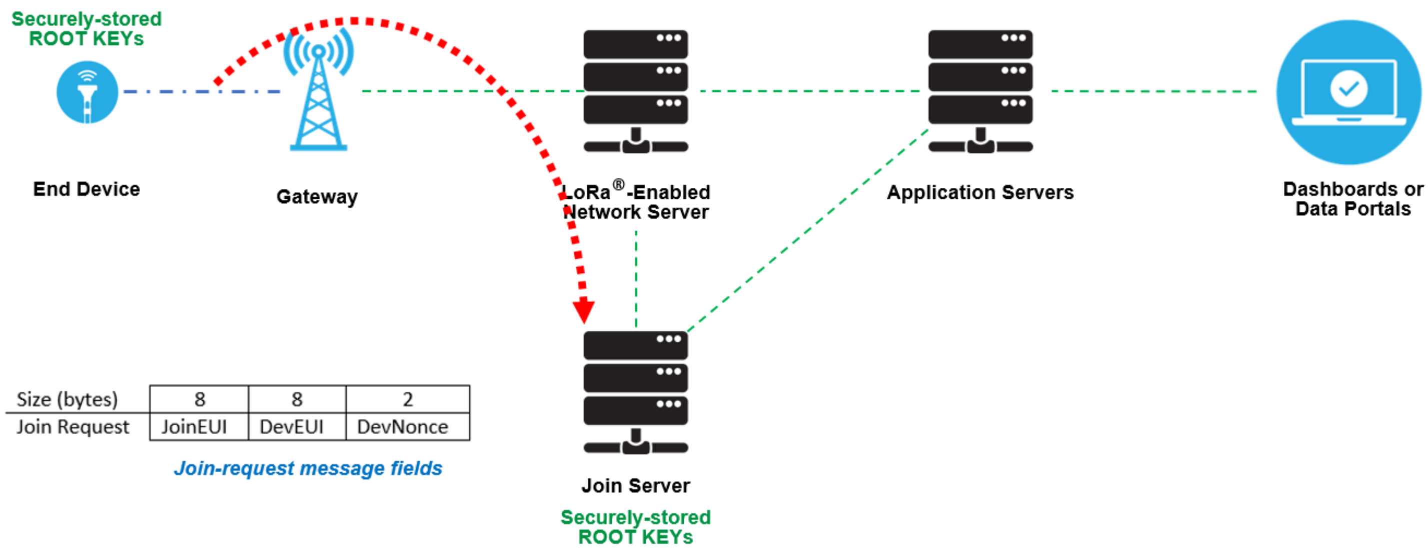



*Figure 19. Sending a join request message to the join server*

After the join server authenticates the device requesting to join the network, it returns a *join accept* message to the device, as illustrated in Figure 20.

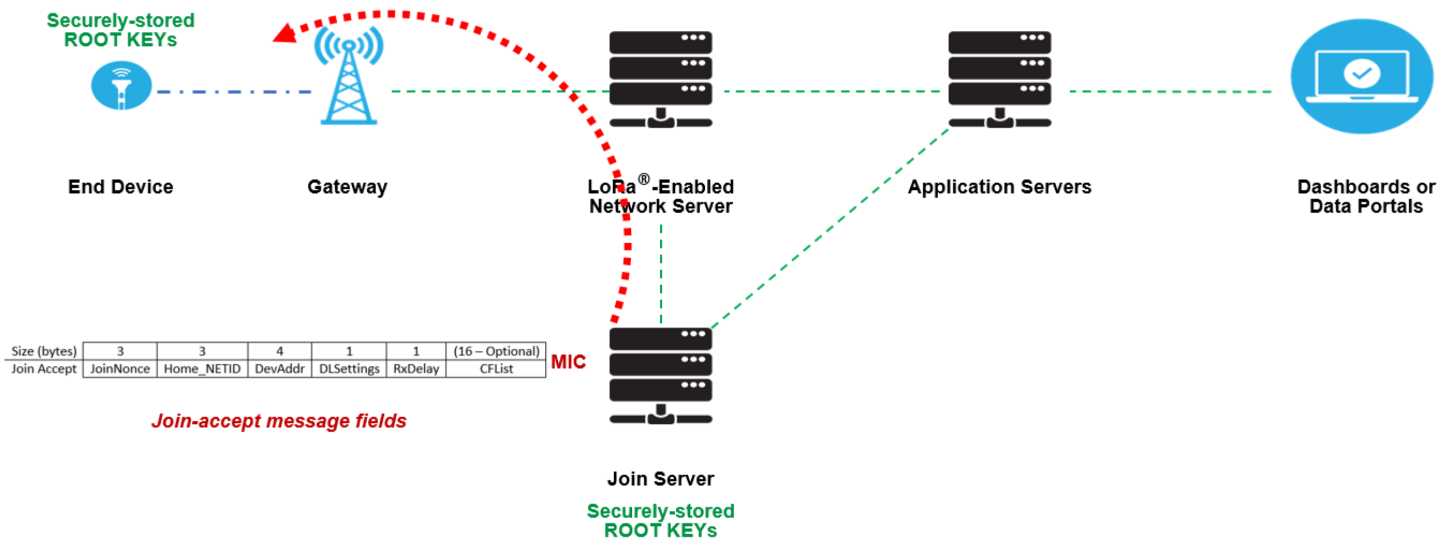



*Figure 20. Sending a join accept message to an end device*

Next, the end device **derives session keys locally**, based on the DevEUI, Join EUI, DevNonce, root keys and fields in the join request and join accept messages. On its end, the join server also derives session keys from the serial IDs, root keys and fields in join requests and join accept messages. Finally, the join server shares session keys with network and application servers, as illustrated in Figure 21.

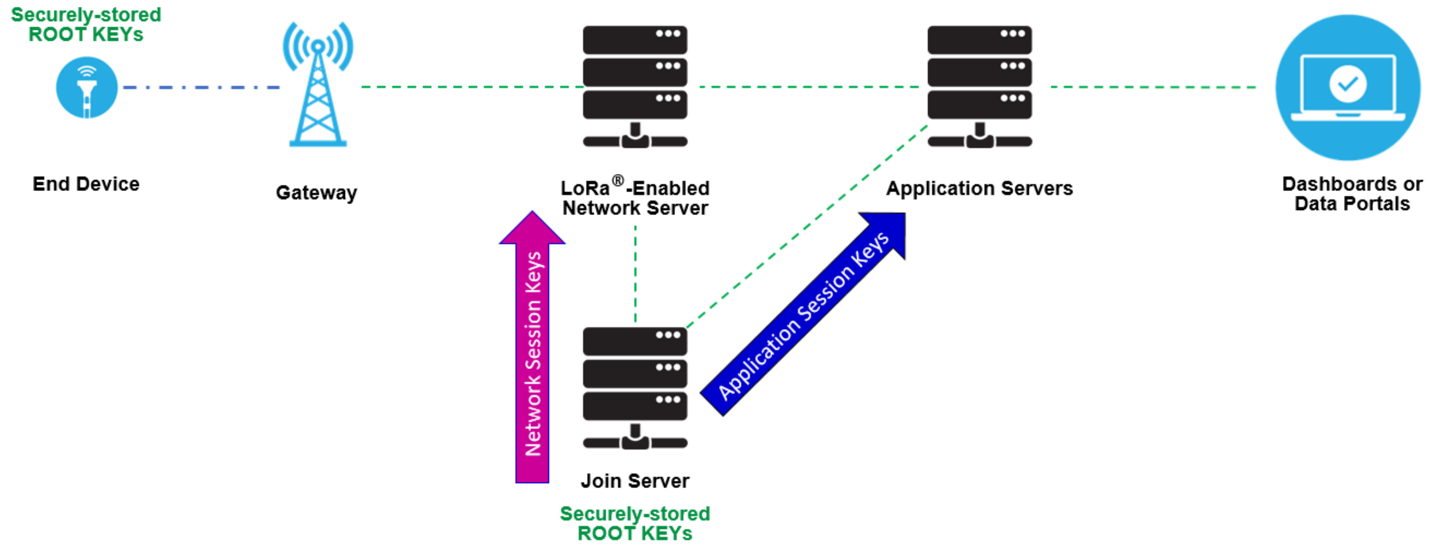



*Figure 21. Session keys are shared with the network server and the application server*

Figure 22 illustrates the security of data packet transmissions. The control traffic between the end device and the network server is secured with a 128-bit AES Network Session Key (NwkSKey). The data traffic that travels between the end device and the application server, is secured with a 128-bit Application Session Key (AppSKey). This method ensures that neither the gateway nor the network server can read the user data.

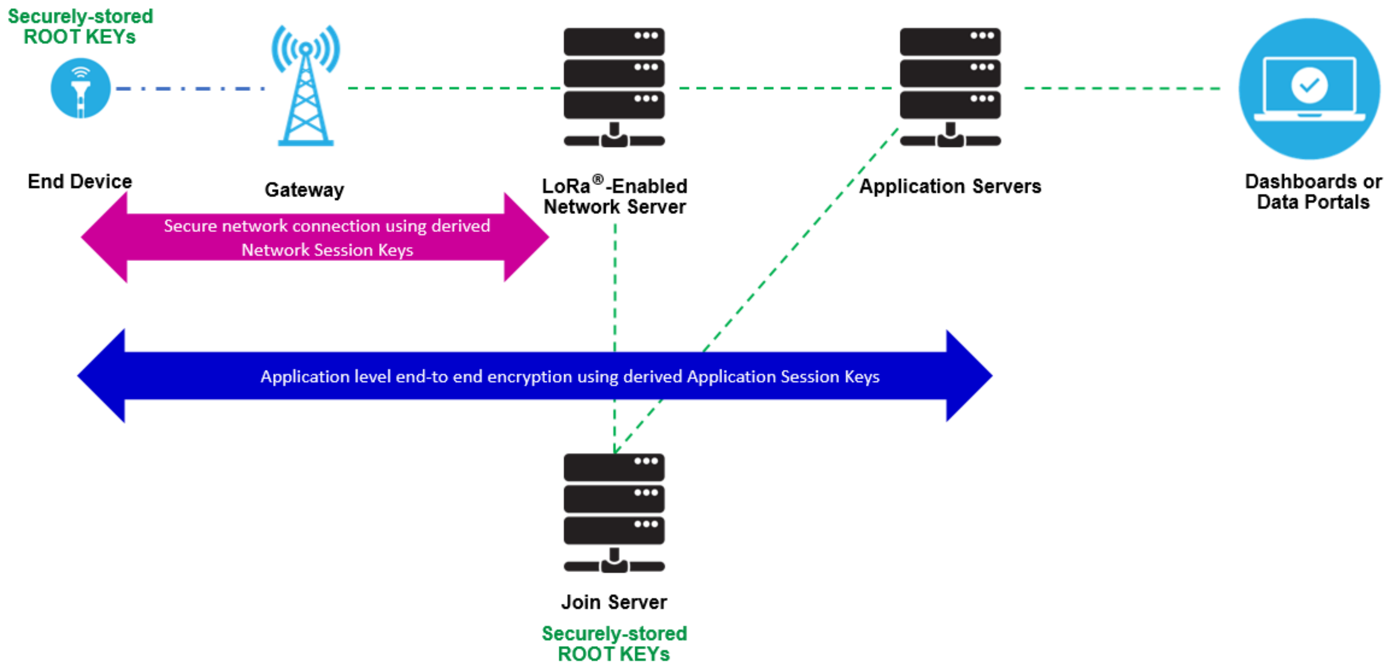



*Figure 22. Secure transmission of data packets*

## 3.4. Device Classes: A, B and C

LoRa-based end devices may operate in one of three modes, depending on their device class. All such devices must support Class A operation. Class B devices must support both Class A and Class B modes, and Class C devices must support Class A mode of operation with Class B mode being optional. These modes of operation have to do with how the devices communicate with the network.

### 3.4.1. Class A Devices

Figure 23 shows how the Class A mode of operation works.

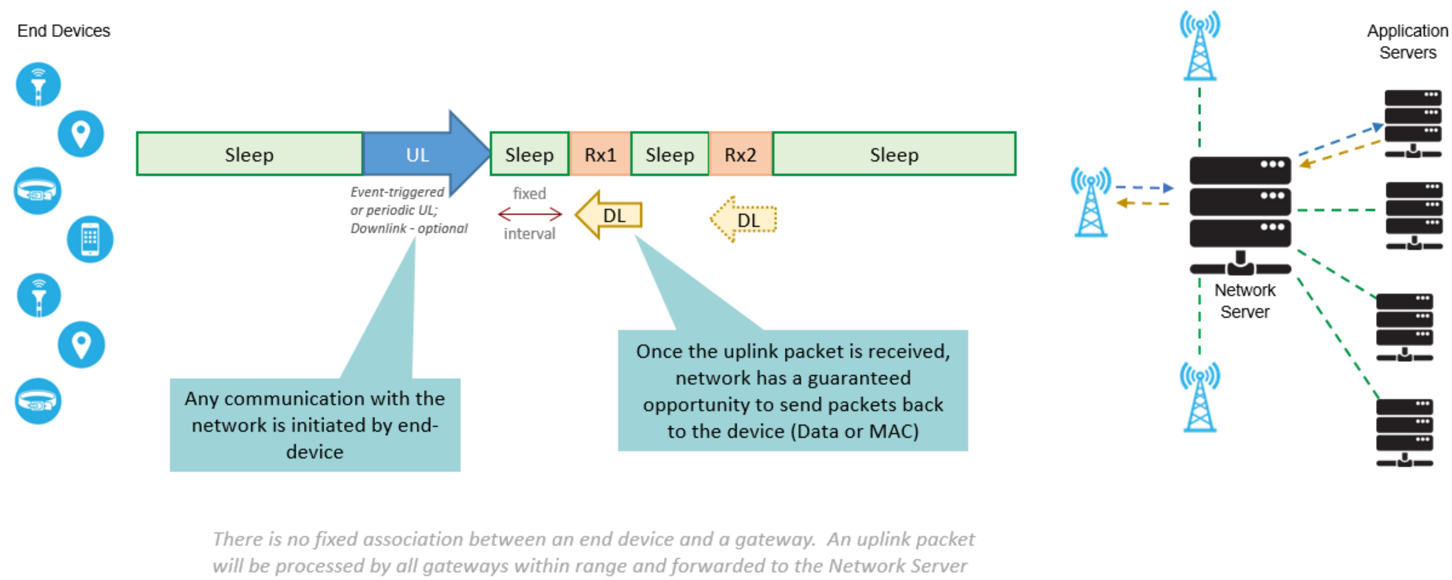



*Figure 23. Class A operation*

In this case, the end device spends most of its time in an idle state, (that is, in sleep mode). When there is a change in the environment related to whatever the device is programmed to monitor, it wakes up and initiates an uplink, transmitting the data about the changed state back to the network (Tx). The device then listens for a response from the network, typically for one second (although this duration is configurable). If it does not receive a downlink during this *receive window* (Rx1), it briefly goes back to sleep, waking a moment later, again listening for a response (Rx2). If no response is received during this second Rx window, the device goes back to sleep until the next time it has data to report. The delay between Rx1 and Rx2 is configured in terms of a delay from the end of the uplink transmission.

| | |
|---|---|
| **NOTE** | There is no way the application of the end device can wake up a Class A device. Given this limitation, Class A devices are generally not suitable for actuators. |

Figure 24, Figure 25 and Figure 26 illustrate these communication patterns.

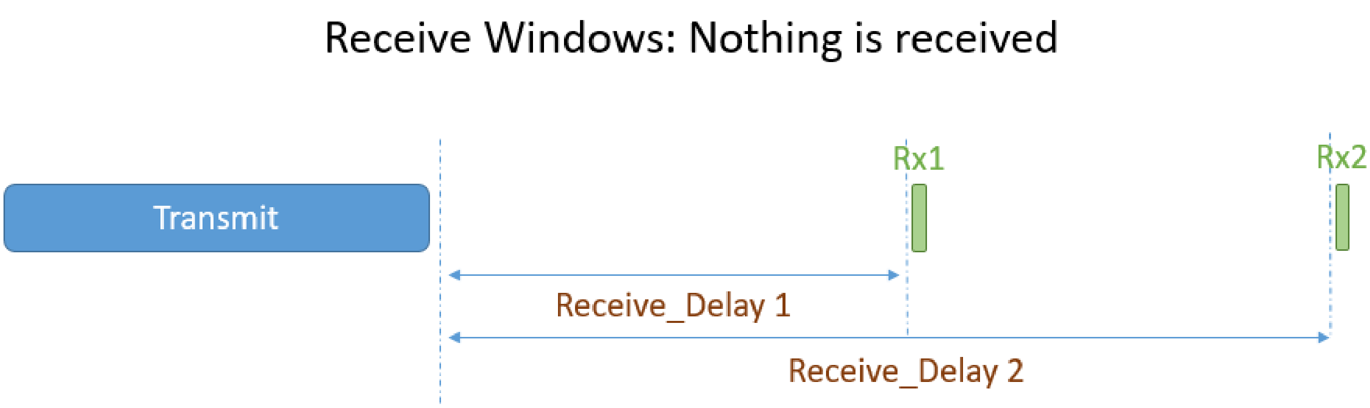



*Figure 24. Class A operation when nothing is received*

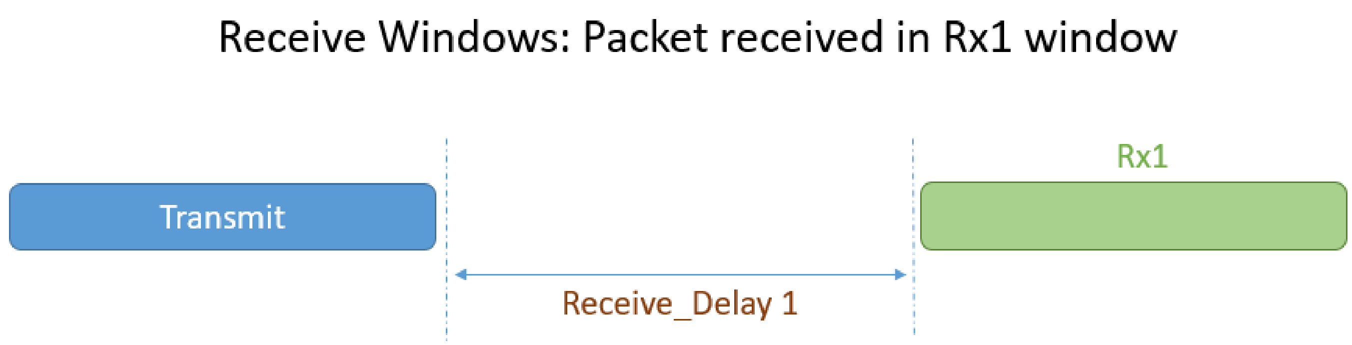



*Figure 25. Class A operation when a data packet is received in the first receive window*

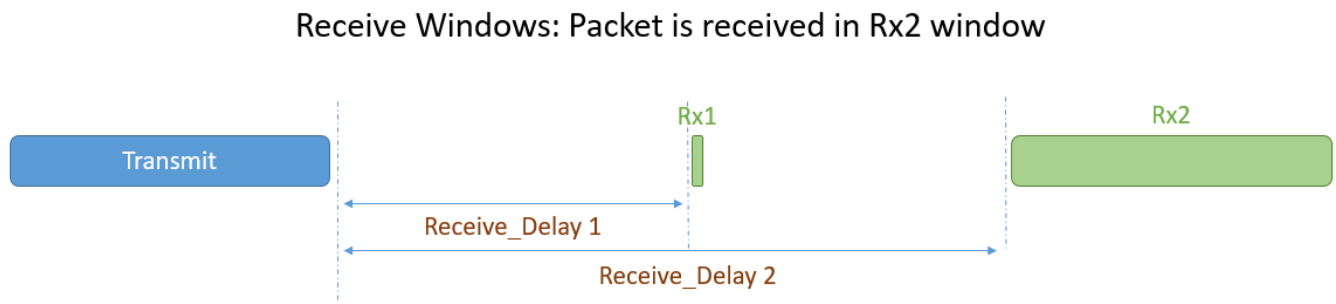



*Figure 26. Class A operation when a data packet is received in the second receive window*

**NOTE**

A device will not try to send another uplink message until either:

1. It has received a downlink message during Rx1, or
2. The second receive window following the last transmission is complete

## 3.4.2. Class B Devices

An enhancement of Class A, LoRaWAN Class B mode offers regularly-scheduled, fixed-time opportunities for an end device to receive downlinks from the network, making Class B end devices suitable for both monitoring sensors as well as actuators. All LoRa-based end devices start in Class A mode; however, devices programmed with a Class B stack during manufacturing may be switched to Class B mode by the application layer.

End devices in Class B mode provide for regularly-scheduled receive windows, in addition to those that open whenever a Class A-style uplink is sent to the server.

### Class B Beacons

For the Class B mode of communication to work, a process called *beaconing* is required. During the beaconing process, a time-synchronized beacon must be broadcast periodically by the network via the gateways, as illustrated in Figure 27. The end device must periodically receive one of these network beacons so that it can align its internal timing reference with the network.

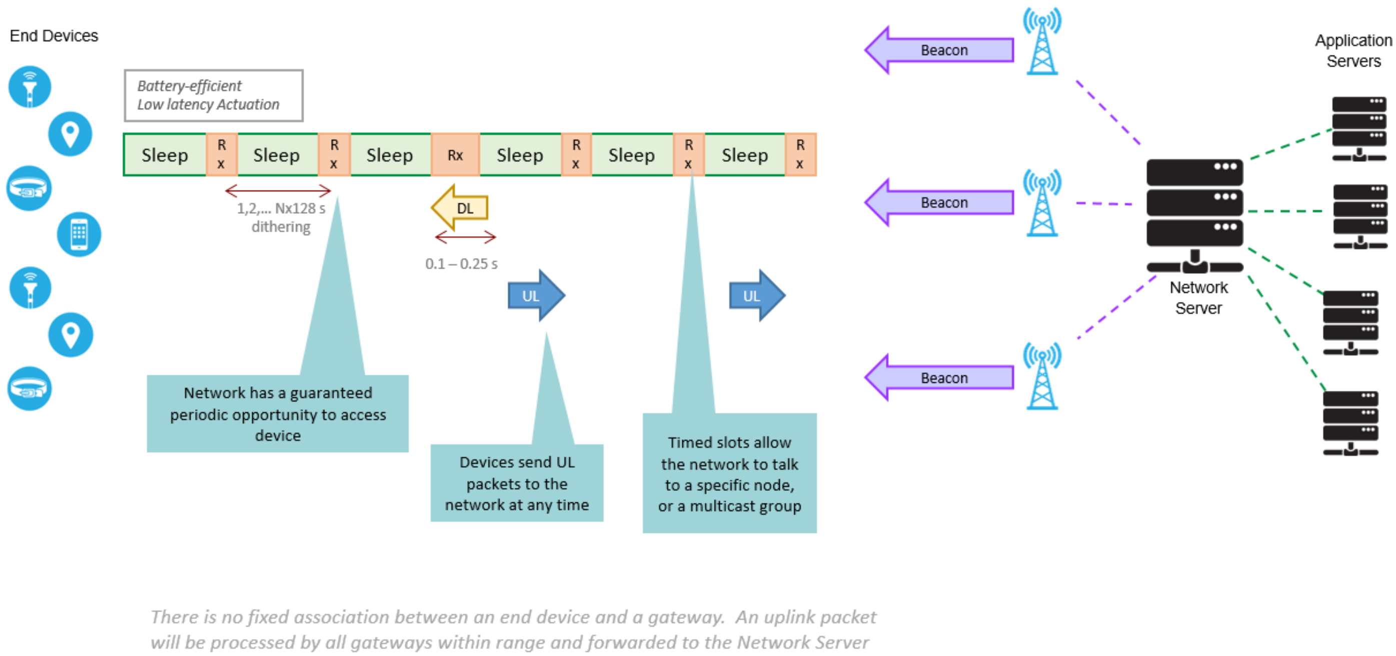



*Figure 27. Class B beaconing operations*

Devices use beacons to derive and align their internal clocks with the network. Devices do not need to process every beacon if the device is already aligned. In most cases, realigning several times a day is sufficient, with a minimal impact on battery life, as illustrated in Figure 28.

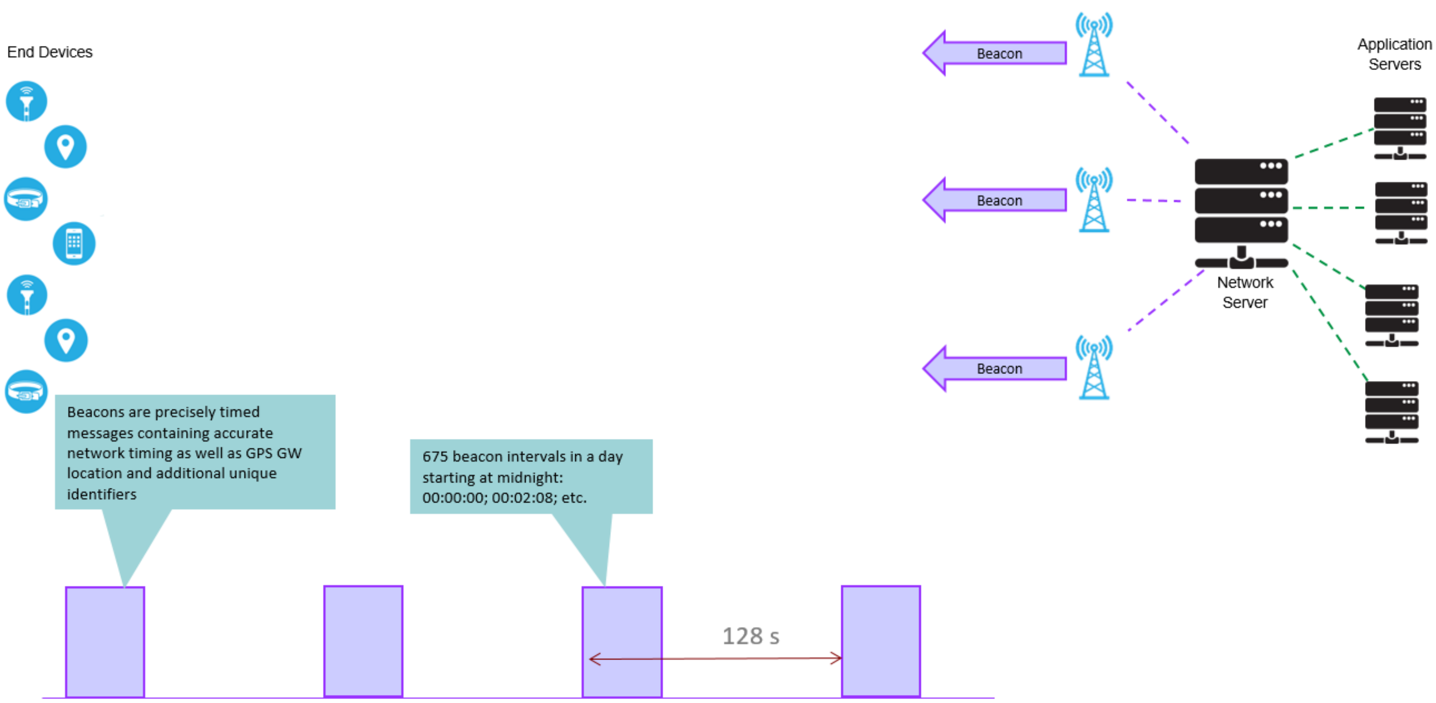



*Figure 28. Periodic Class B beaconing for device synchronization*

Based on the beacon's timing reference, end devices can open receive windows (*ping slots*) periodically. Any of these ping slots may be used by the network infrastructure to initiate a downlink communication, as shown in Figure 29. In order for a LoRaWAN network to support Class B devices, all the LoRaWAN gateways in this network need to have a built-in GPS timing source, so they all can be synchronized to the exact beacon timing.

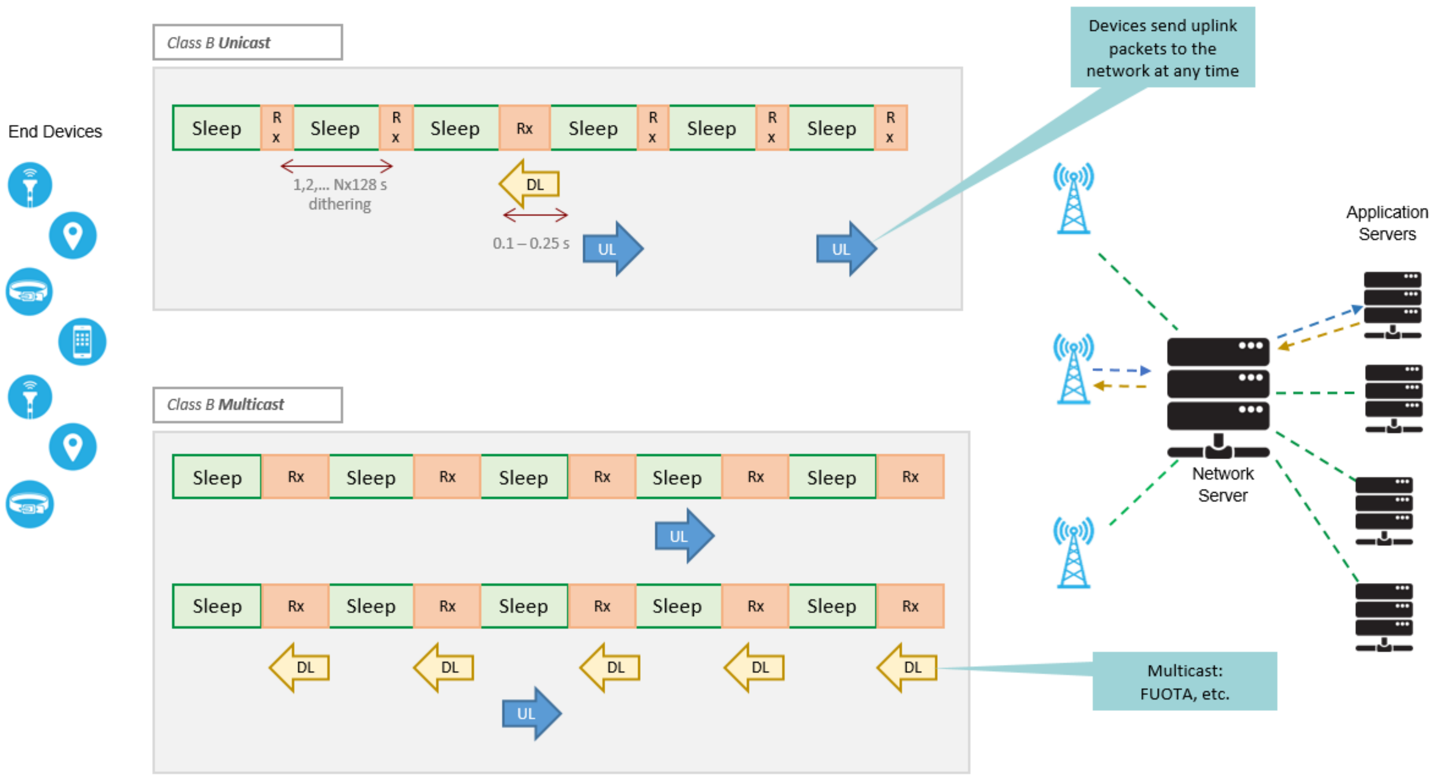



*Figure 29. Class B ping slots*

**NOTE** | Class B devices can also operate in Class A mode.

## 3.4.3. Class C Devices

Class C devices are always "on"; that is, they do not depend on battery power. Class C devices include such things as street lights, electrical meters etc. These devices are always listening for downlink messages, unless they are transmitting an uplink. As a result, they offer the lowest latency for communication from the server to an end device.

Class C end devices implement the same two receive windows as Class A devices, but they do not close the Rx2 window until they send the next transmission back to the server. Therefore, they can receive a downlink in the Rx2 window at almost any time. A short window at the Rx2 frequency and data rate is also opened between the end of the transmission and the beginning of the Rx1 receive window, as illustrated in Figure 30.

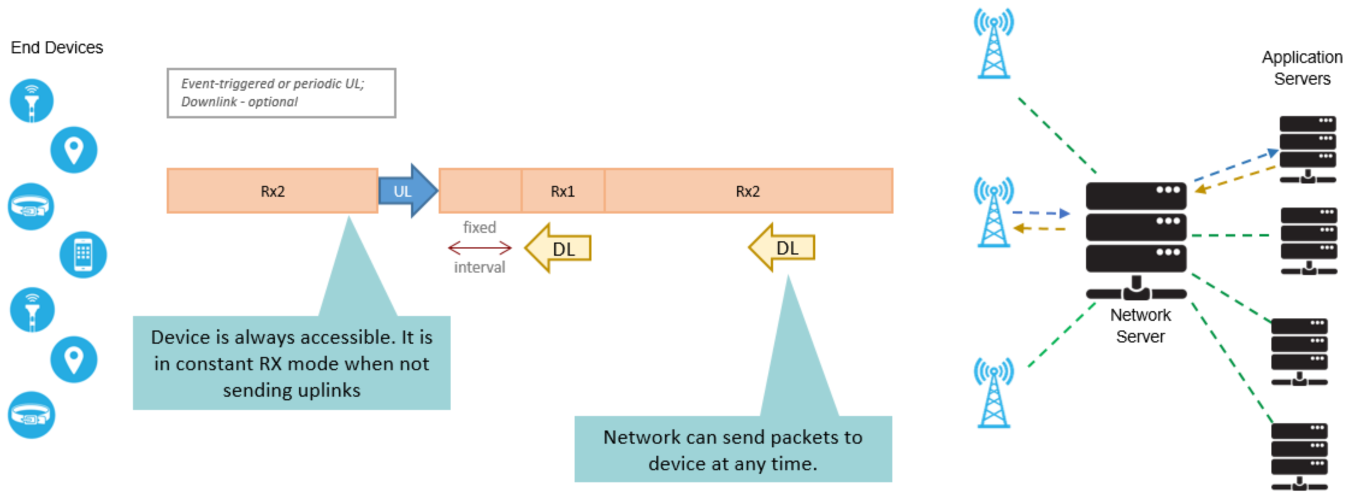



*Figure 30. Class C operation*

# Chapter 4. The LoRa Alliance

With more than 500 member companies, the LoRa Alliance is one of the fastest-growing technology alliances. A community of innovators, the LoRa Alliance is committed to standardizing low power wide area networks (LPWANs). To this end, the group provides the LoRaWAN Specification (https://lora-alliance.org/search/specification) free of charge. The specification is based on open standards and provides for certified interoperability.

The LoRa Alliance also offers the LoRaWAN Certification Test Tool, to help manufacturers ensure that their devices are fully LoRaWAN-compatible prior to sending those devices to an Authorized Test House for formal LoRaWAN Certification testing.

For more information on the LoRa Alliance, visit their website: https://lora-alliance.org.

# Disclaimer

**IMPORTANT**

Information relating to this product and the application or design described herein is believed to be reliable, however such information is provided as a guide only and Semtech assumes no liability for any errors in this document, or for the application or design described herein.

Semtech reserves the right to make changes to the product or this document at any time without notice. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. Semtech warrants performance of its products to the specifications applicable at the time of sale, and all sales are made in accordance with Semtech's standard terms and conditions of sale.

SEMTECH PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS, OR IN NUCLEAR APPLICATIONS IN WHICH THE FAILURE COULD BE REASONABLY EXPECTED TO RESULT IN PERSONAL INJURY, LOSS OF LIFE OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE. INCLUSION OF SEMTECH PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE UNDERTAKEN SOLELY AT THE CUSTOMER'S OWN RISK. Should a customer purchase or use Semtech products for any such unauthorized application, the customer shall indemnify and hold Semtech and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs damages and attorney fees which could arise.

The Semtech name and logo are registered trademarks of the Semtech Corporation. All other trademarks and trade names mentioned may be marks and names of Semtech or their respective companies. Semtech reserves the right to make changes to, or discontinue any products described in this document without further notice. Semtech makes no warranty, representation or guarantee, express or implied, regarding the suitability of its products for any particular purpose. 

# Wi-Fi® mesh networks: Discover new wireless paths

TEXAS INSTRUMENTS

**Victor Asovsky**
*WiLink™ 8 System Engineer*

**Yaniv Machani**
*WiLink 8 Software Team Leader*

*Texas Instruments*

# Table of Contents

# Abstract

With recent advances of in Wi-Fi® technology, specifically in its ease of implementation and ability to meet the requirements of a majority of connectivity use cases; it has become one of the dominant wireless solutions. The continuous progress in this wireless technology has propelled its successful penetration in a variety of applications. So, it also became a highly requested connectivity solution in large-scale use cases for which it was not originally developed.

The wireless networks became more complicated with sophisticated structures in which the regular access point (AP)-based topology appeared to be a less efficient solution. The requirement of constructing such networks in which more than one hop will be used, has enforced the IEEE organization to compose the IEEE Std 802.11s-2011 mesh networking amendment on top of the IEEE Std 802.11™ specifications document. This amendment became an integral part of the IEEE Std 802.11-2012.

In this paper, we will discuss wireless mesh networking from the top down. We will cover its main concepts and advantages and show a variety of possible wireless mesh applications along with a general deployment consideration. We will then deep-dive into the core features composing its implementation and, finally, show what improvements Texas Instruments (TI) has released based on the Linux® open-source 802.11s implementation.


# Introduction

It's possible now to have a fiber optic cable connected directly to an expensive, high-end router, but it will just make it more frustrating when the Wi-Fi signal is not good enough to hold a solid connection to your device. Wi-Fi extenders and repeaters can be used to extend range, but due to the nature of the wireless medium, in many cases such installation will fail to provide the desired bandwidth.

Furthermore, it is also required to pre-define roles and responsibilities for each device. The access point will have to be placed in locations that will permit providing service to all of its clients and this fixed positioning has to be maintained to preserve the network connectivity.

Wi-Fi mesh networks have disrupted these two basic assumptions. Every mesh node is equivalent to the others, which makes the network independent with a self-forming capability. After an initial setup, the device can be located wherever it's needed; it can act as an endpoint, a range extender and even a gateway to an external network. A mesh network can rearrange itself using a self-healing mechanism, to support cases in which some nodes have changed their position or have been turned off.

With such capabilities, new use cases and topologies for Wi-Fi technology usage have been opened. These use cases range from a closed loop system that supports machine-to-machine (M2M) topology, through a home automation deployment and up to a smart grid use case that can be constructed using a huge number of hops and nodes.

The mesh-capable device might have only basic IEEE 802.11s features and be able to construct a mesh network or be loaded with more sophisticated features like precise synchronization between devices, a concurrent operation of the mesh and other Wi-Fi roles, a dynamic transition between mesh network and AP-based network and so on.

However, mesh networks have a major consideration on the bandwidth they can handle. As the best path to transfer data from one point to the other can constantly change, it is crucial to determine the best route in a dynamic, reliable way and as quickly as possible. Although, there are many routing algorithms out there, it's not a trivial task to fine tune the best path selection algorithms to achieve an optimal Wi-Fi behavior.

# Mesh network use cases

## General capabilities

A mesh network can be used for a wide variety of scenarios, outside of the basic capability to transfer data over a distributed deployment. There are several advanced capabilities that might be required from mesh devices in order to support most of the use cases. Examples of such capabilities can be:

- Ethernet bridged mesh station (STA) is a simple and robust way of creating a mesh network leveraging its benefits without overloading any existing infrastructure and even taking advantage of a separated channel. When reliable and high-performance data transfer is required, this should be the selected configuration. A great example of this is wireless audio speakers where the music is played across multiple devices, sometimes out of range of the home AP.
- Concurrent operation of mesh STA and AP role can be used to allow connection of legacy station devices to a mesh network. This can be very useful in M2M scenarios where the devices are all connected to each other through the mesh network but can be controlled and accessed through STA devices handled by the operators.
- Concurrent operation of mesh STA and STA role can be used to connect the mesh network to a standard AP infrastructure for Internet access; it is a huge advantage for home automation devices. In such a configuration, all of the appliances will connect to each other seamlessly and will form an independent network which can be used to pass information between them while reducing the dependency on the home AP. A single device connected to the home AP will provide Internet access when needed.
- Dynamic transition between STA and mesh STA link establishment. In this type of configuration there will be just a single active link the majority of the time, as opposed to the case where both links are kept concurrently. The transition will be based on the device's needs. If there is a good link quality with the AP, the STA link will be kept, allowing the use of the legacy power-save schemes. Once the link with the AP is degraded, the device will use the mesh STA link for improving its performance.
- An extended feature is precise synchronization between mesh devices. For STA devices this is a trivial and a required capability, by the IEEE 802.11 specification, to allow AP beacon tracking, power-save-mode operation and more. For mesh STA, this is not mandatory but it can be very helpful when a master clock is required to align all of the devices. Such a

feature can be used when synchronized music is played between speakers in the same room.

The above advanced capabilities have two common key advantages which translate into the main mesh use cases. They will be listed in the following sections.

## Range extension use case

One of the major benefits of mesh deployment is the possibility to pass data between two devices that are not in range of each other. This benefit is created by taking advantage of the connection to the mesh STA devices in between. Devices can be connected in a row, one after the other, forming a chain of nodes (somewhat like a chain of city street lights). Since each added hop has an impact on the throughput and the latency of the transferred data, the number of links in such a chain and the location of the external network connection should be adjusted to meet the expected network performance.

A classic "chain" example can be city lights, connected to one or more external network APs, allowing wireless data transfer, status reports and remote control using standard Internet protocol (IP) infrastructure.

Another common deployment example can be the wireless speakers scenario where there is one speaker connected to an Internet music streaming service and providing that connection to all other speakers in that network. A mesh network can easily double the range of the Internet access and enable dynamic positioning of speakers around the house.

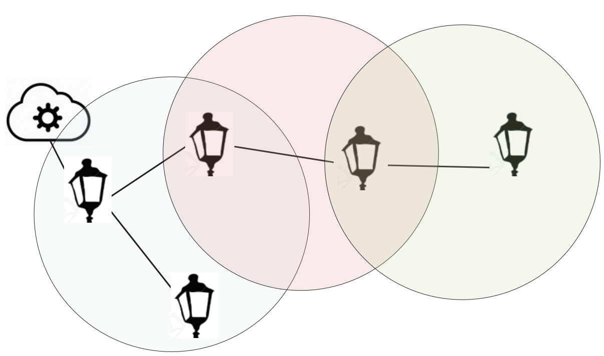


*Figure 1: Use case. Range extension*

## AP offloading use case

The demand for wireless bandwidth within a specific network is constantly increasing; more and more mobile devices are streaming video and audio content with high-quality requirements. Using a smart mesh deployment can actually offload some of the load from the legacy AP device by enabling a direct data transfer between the devices.

One of the most common examples is the wireless audio speakers scenario.

Today, many audio services can support a multi-speaker streaming playback. The audio source can be an Internet music streaming service, such as Spotify®, Pandora®, TuneIn, etc., but also local content played through Digital Living Network Alliance (DLNA) services. In any of these cases one of the devices is selected to be the "master" of the network and it is the one supplying the content to the rest of the speakers. With the legacy AP-based topology, all the information has to go through the AP to get to the other devices, as demonstrated in Figure 2 below.

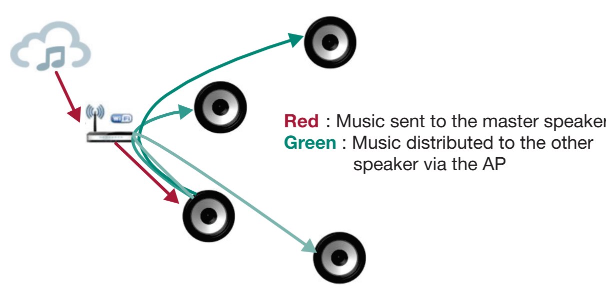



*Figure 2: AP standard usage*

Using the mesh network can significantly reduce the load on the AP and help utilize the air medium in

higher efficiency as the speakers can transfer data directly from one to the other, as shown in Figure 3.

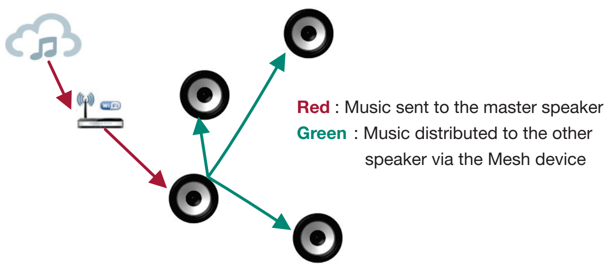



*Figure 3: Use case. AP offloading*

# Wi-Fi mesh key features

## Homogenous

Wireless mesh networks inherit capabilities from legacy AP/STA modes. Each mesh device acts as an autonomous basic service set (BSS) which is reflected in the beacons' transmission, connection handling and so on. There are no pre-configured roles to the nodes, which makes the network highly flexible and simplifies the deployment.

## Self-forming

The wireless local area network (WLAN) mesh is defined as a self-forming network. The detection of other mesh devices is done by listening on a selected channel for the presence of other devices with mesh dedicated information elements (IEs) in its beacons. A suitable mesh device is one that has the same network name and uses the same security protocol. Once the suitable network is detected, a connection will be automatically initiated. The connection can be open or closed (secured) using a pre-shared key.

## Dynamic path selection

One of the key concepts of a distributed network is redundancy of links in the system. This redundancy allows better resilience to node failures and allows multiple options to reach from one point to the other. Together, it imposes the challenge of selecting the best path to the desired destination.

## Process description

A mesh station uses a built-in algorithm to compute a path cost, or metric. For each destination, it will be required to calculate the optimal path with the lowest link cost/metric. This is done by incorporating the hop count, signal quality, data rate and more.

The path selection process is invoked periodically, even if the mesh network is constructed as a stationary network. This is required since the air conditions might change, a device might change its position or disappear and the connection quality between devices that currently construct the best path might become worse (see Figure 4).

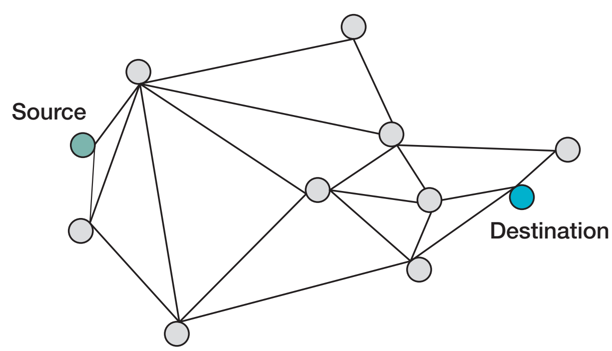



*Figure 4: Only one optimal path between source and destination*

The path selection algorithm selected by the IEEE to be incorporated into the specification is hybrid wireless mesh protocol (HWMP). It is divided into two main modes: "on-demand" and "proactive", to meet the needs of different mesh network topologies and use cases.

## "Proactive" mode

In this mode, once one or more devices are defined as a root mesh device through a "root announcement" frame, all the other mesh devices will start to form a path towards that node. The path selection process will be periodically invoked even when there is no data to be sent. Due to the large amount of packets continuously transmitted in this mode, it can impact utilization of networks with a high number of nodes and thus it's not recommended. Figure 5 shows the process:

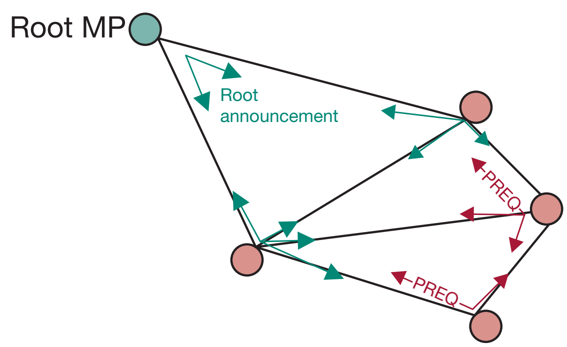



*Figure 5: Root announcements distributed, causing PREQ to be sent*

## "On-demand" mode

The "on-demand" path selection mode is the default operation mode in the 802.11s open-source solution since it meets the needs of most of the scenarios and topologies. Also, it improves power consumption and network utilization since it will be triggered only when data is required to be sent:

- Source have data to send.
- Broadcast path request (PREQ) is sent from the source.
- Broadcast PREQ is propagated through the network nodes.
- Destination receives all the PREQ frames, and replies with a unicast path response (PREP) to the device constructing the best path to the source.
- PREP is forwarded by intermediate nodes until it reaches the source.

What might be counterintuitive is that the responsibility for determining the best path is made by the destination mesh STA and not by the originator, while the other nodes along the best path are used for passing the PREP frame from the destination toward the data source.

Figure 6 shows the process:

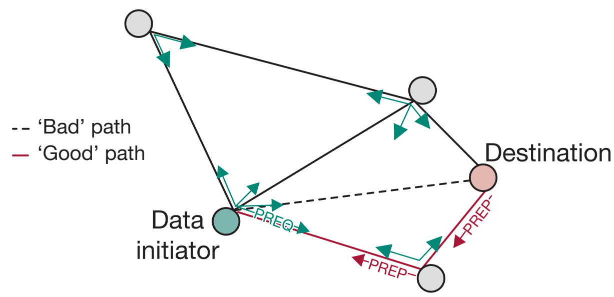



*Figure 6: PREQ frames are distributed over the network reaching to the destination. The destination replies using the best path.*

Intermediate nodes that pass the PREP will also capture the path cost to adjacent nodes, this is valuable to allow sharing the metric information and reduce the packet distribution. On the other hand it can cause these nodes to capture the non-optimal path and result in a performance impact.

## Self-healing

The self-healing mechanism is responsible for detecting mesh devices gracefully leaving the network or an abrupt disappearance and takes an immediate action for its fixing.

In both cases, once identified, the data source will have to construct a path toward the destination. The time for the whole process has to be quick to prevent data loss and high latency. A good response time is considered within the range of 0.5–2 sec.

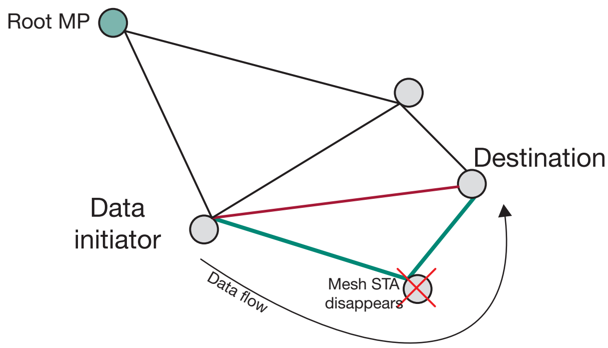



*Figure 7: Mesh network where mesh device disappears*

## Possible issues in mesh networking

### *Path metric*

The air metric calculation described in the specification is based on the effective transmission rate and the packet error rate. In many cases, mesh devices don't have or have a non-updated rate's information. As the result, it will not reflect the real current state and a wrong path will be selected. In fact, in any case where the data initiator has a unidirectional data sent, like User Datagram Protocol (UDP), intermediate nodes will not have any reliable information on the link quality and the optimal path will not be selected.

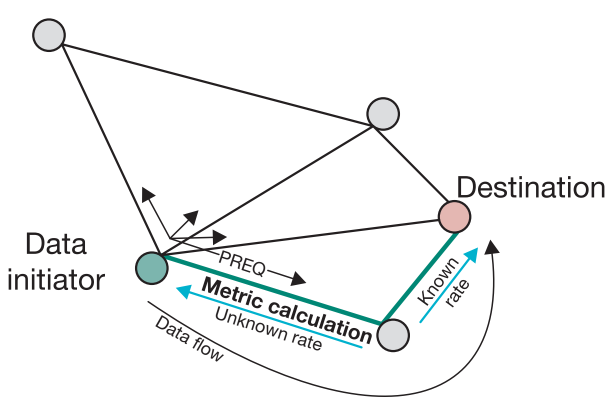



*Figure 8: Path selection. Metric calculation*

### *Path selection process' reliability*

As described above, the path selection process starts with a broadcast transmission of a PREQ frame. In general, broadcast transmission is not reliable as unicast (to a specific destination) as it doesn't require an acknowledgement and it doesn't have any retry mechanism. As a result, reception of packets at the destination is not guaranteed and particularly in the mesh device that constructs the next hop of the preferred path. Moreover, since the broadcast frame is forwarded by all mesh devices until reaching the destination, its propagation might be broken over the better path at any node and be successful over some path with poor conditions.

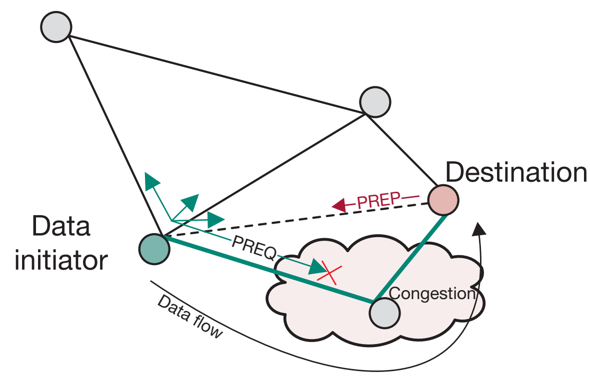



*Figure 9: PREQ lost, PREP returns on the non-optimal path*

The reason for such an incident can easily be explained. Figure 9 shows the data flow over the best path, which means that the air around these mesh devices is congested with the WLAN transmissions. When the path selections start, broadcast PREQ frame transmission has, as any other frame, some probability to get lost due to collision. In this case, PREP will be sent over the poor link, the path will be switched and the performance will be impacted. In the next path selection cycle, figuring the optimal path will be attempted again, it's quite possible that such a phenomena will occur again while fixating the derogated path.

# WLAN mesh deployment considerations

When deciding to use a WLAN mesh solution, additional considerations should be taken into account. Those parameters will also have an impact on the overall systems behavior. A few of them are described below and should be also considered during network's planning.

## Number of hops

Let's consider a scenario of a network that is contracted as the chain of devices—it should be noted that such topology might have some limitations. The limitations are reflected in the network's performance in terms of bandwidth and delay. Theoretically, this chain might be unlimited in the number of hops through it; however the data's propagation through such a network might have a very large delay accumulated when moving from device to device. Also, the maximum data throughput that will be able to flow through a large number of hops will be limited due to the air access process in the wireless network. Adding intermediate points defined as gateways, to permit data's exit though will not constrain the data to propagate from edge to edge or through a large number of hops.

## Number of devices

It is known that the total bandwidth for the WLAN operation has some physical limitation value. This maximal bandwidth will be divided between all devices connected on the same channel and within the same range. Mesh flexible positioning and configuration might come in handy in cases where there is a need for a large number of devices. Dividing the network into clusters and taking leverage of the mesh range extension capabilities will increase network usability and overall performance.

## Hidden nodes

When talking about large-scale or long-range networks, a key aspect of hidden nodes must be taken into consideration. The hidden node term refers to the situation when devices don't or almost don't hear each other, but still operate on the same channel and contest on access to the air. In an AP-based network, such a process is easier to control, due to the centralized nature of the network and a dedicated protection mechanism. Contrary, in a mesh network, the same protection mechanism will not always resolve the hidden nodes situation. So, network designers need to be aware of such wireless-based network problems and design the network properly.

## Connection radius

The wireless network permits connections between devices even if they are located at the edge of the lowest rate sensitivity level. Such a connected device will be able to operate most of the time at the lowest modulation rates, which requires a long air time allocation and, as a result, significantly decreases bandwidth. This option is essential when this is an only way to permit some device to join the network, but can be devastating in the network that doesn't require such rescue state and might propose an alternative solution. In mesh networks, optimizing the sensitivity threshold will permit the network to operate effectively in terms of rates' usage and limit the number of connected devices to one device, hence will scatter the network's load and enlarge the total bandwidth (see Figure 10 on the following page).

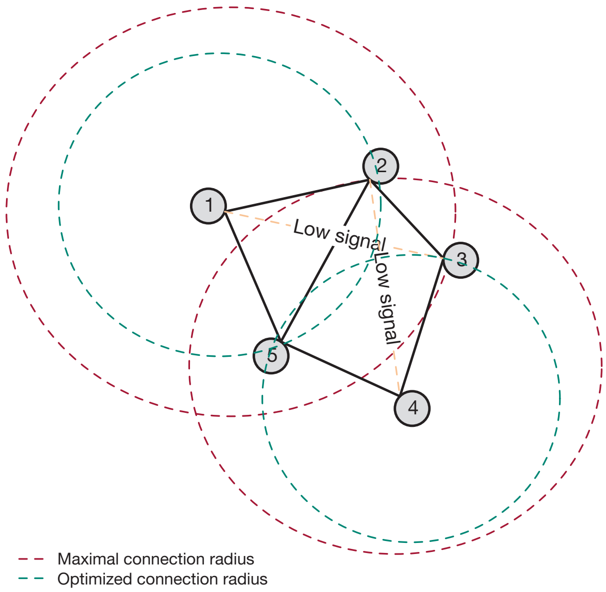



*Figure 10: Network's maximal and optimized connection radius*

# The Wi-Fi mesh solution from Texas Instruments

The Texas Instruments WiLink™ 8 combo-connectivity device has an integrated solution for the wireless mesh network, using Wi-Fi technology. It is based on the IEEE 802.11s, MAC80211 open-source implementation. It permits a dynamic range of configurations, standalone and combined operation with other modes. Most importantly, it provides a reliable and robust implementation that overcomes many of the challenges described earlier to allow high-quality products to leverage this technology.

Although the basic 802.11s open-source implementation has been available for a few years, it has not dealt with many challenges when trying to achieve stable, high-throughput results. TI's WiLink 8 Wi-Fi-based mesh network solution enables developers to better overcome these challenges.

## Path selection

As stated before, path selection is the core algorithm of the mesh network. It can make the network performance great or unusable. TI's WiLink 8 software implementation uses the "on-demand" mode as it meets the needs of most real-life scenarios. It kept its basic concepts that allows better power consumption, air utilization and dynamic deployment, but handled the reliability downfalls it has. The WiLink 8 solution will consistently find the best and the most stable path toward the destination on every path selection cycle in regular conditions. One other key improvement is adjusting the air link metric calculation also in case of obscurity of the effective rate toward the mesh device to which this calculation should be invoked as described in the section entitled "Path metric".

The example described below compares the data flow in the legacy 11n connection versus three WiLink 8 devices forming a mesh network. In both cases, the destination device is getting further away from the data source. Such a test is usually referred to as RvR (rate vs. range), where range is simulated using a variable attenuator between the nodes. In a mesh network the variable attenuator is changed linearly causing natural rate degradation and triggering the path selection protocol to alternate the data flow to the best path.

Figure 11 shows the setup and the data flow when the variable attenuator changes across the direct link and then across the second hop link.

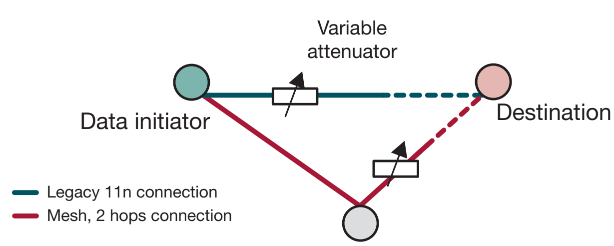



*Figure 11: Basic path selection scenario*

Figure 12 shows the rate selection in the different attenuation. It can be clearly seen that on the 89dB mark, a new path is selected; stopping the rate drop and allowing a stable data path for a larger range.

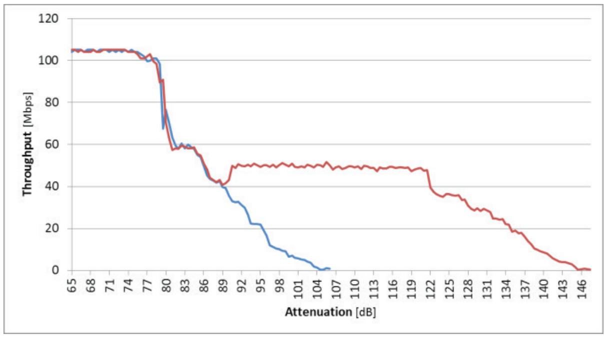



*Figure 12: Mesh path selection influence on a system's performance*

Referring to the reliability problems described in the section entitled "Path selection process' reliability" from which the path selection process might suffer and the wrong decision may be taken, the WiLink 8 solution provides a highly-reliable path selection process, better propagation over the network and the final correct decision also in a vulnerable network.

The following basic mesh network topology in Figure 13 describes the setup for path selection reliability analysis.

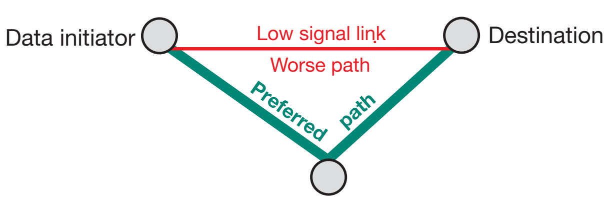



*Figure 13: Path selection reliability setup*

Figure 14 shows the path selection reliable behavior reflected in the stable data flow over the Wi-Fi-based mesh network.

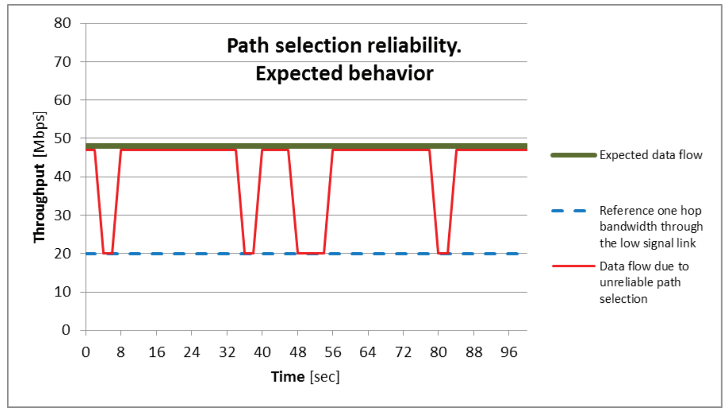



*Figure 14: Path selection reliability expected behavior*

Figure 15 shows the real result tested for the scenario described in Figure 13. It compares the actual throughput result with the stable path selection decision to use a two-hops path versus a pre-measured throughput as if only the direct path had been existing.

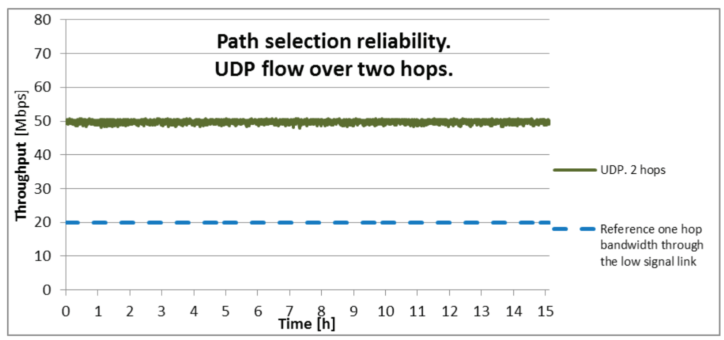



*Figure 15: Path selection reliability. Actual result for the data flow over two hops for 15 hours*

## Bandwidth example for multiple number of hops in the mesh network

To emphasize the importance of choosing the correct Wi-Fi-based network type, an example of the mesh network, including targeted use case needs, with a multi-hop topology is presented in Figure 16 on the following page.

This example shows that, due to the inherent Wi-Fi-based network nature of air access and the bandwidth share, the network performance

decreases significantly in the first hops but stabilized on the following hops keeping a bandwidth that will suit many applications.

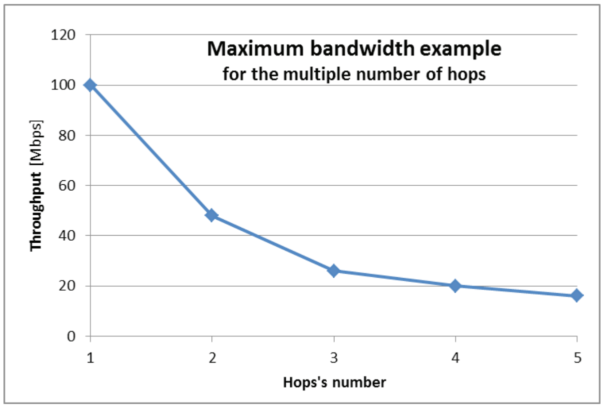



*Figure 16: Bandwidth example over a multiple number of hops*

# Summation

There is no doubt that mesh networks are going to disrupt the way we think about Wi-Fi connectivity. Its extended range, flexibility and better network utilization is opening the door for a wide range of new applications and an improved user experience. Texas Instruments is proud to be at the forefront of this movement by providing an open-source Linux solution that is robust, reliable and easy to integrate on both new and existing WiLink 8-based solutions.

For more information on TI's Wi-Fi mesh solution, please visit: **www.ti.com/wilinkmesh**

# References

[i] IEEE 802.11-2012 specification document section 13.10, p. 1382.

[ii] IEEE 802.11-2012 specification document, section 13.9 p. 1381.

## IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as “components”) are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI’s terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

TI assumes no liability for applications assistance or the design of Buyers’ products. Buyers are responsible for their products and applications using TI components. To minimize the risks associated with Buyers’ products and applications, Buyers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of significant portions of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI components or services with statements different from or beyond the parameters stated by TI for that component or service voids all express and any implied warranties for the associated TI component or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI’s goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or “enhanced plastic” are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components as meeting ISO/TS16949 requirements, mainly for automotive use. In any case of use of non-designated products, TI will not be responsible for any failure to meet ISO/TS16949.

| **Products** | | **Applications** | |
|---|---|---|---|
| Audio | www.ti.com/audio | Automotive and Transportation | www.ti.com/automotive |
| Amplifiers | amplifier.ti.com | Communications and Telecom | www.ti.com/communications |
| Data Converters | dataconverter.ti.com | Computers and Peripherals | www.ti.com/computers |
| DLP® Products | www.dlp.com | Consumer Electronics | www.ti.com/consumer-apps |
| DSP | dsp.ti.com | Energy and Lighting | www.ti.com/energy |
| Clocks and Timers | www.ti.com/clocks | Industrial | www.ti.com/industrial |
| Interface | interface.ti.com | Medical | www.ti.com/medical |
| Logic | logic.ti.com | Security | www.ti.com/security |
| Power Mgmt | power.ti.com | Space, Avionics and Defense | www.ti.com/space-avionics-defense |
| Microcontrollers | microcontroller.ti.com | Video and Imaging | www.ti.com/video |
| RFID | www.ti-rfid.com | | |
| OMAP Applications Processors | www.ti.com/omap | **TI E2E Community** | e2e.ti.com |
| Wireless Connectivity | www.ti.com/wirelessconnectivity | | |

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

TEXAS INSTRUMENTS

*Application Report*
*SLLA442–February 2019*

# *CAN and LIN transceiver low-power modes*

*Eric Hackett*

**ABSTRACT**

This application report presents the low-power modes of Texas Instruments's LIN and CAN transceivers, and explains how they work. The need for low-power modes in CAN and LIN applications is explained, how the modes are used in each application, and why they are beneficial to that application. The report ends with a list of LIN and CAN transceivers in TI's portfolio that have low-power modes implemented.

**Contents**



**List of Figures**

**List of Tables**

## Trademarks

All trademarks are the property of their respective owners.

## 1 Introduction

As cars become more intelligent, the amount of power consumed from the battery grows due to the increasing number of electronic circuits implemented. Since the car runs on a finite battery supply, system architects need a way to add more functionality without adding significant drain on the battery. Electronics developers are answering this call by developing devices that included low-power modes. These devices allow certain parts of the system to be switched to consume less power, instead of going through a full power cycle, slowing down response times, or leaving every system on in full power mode.

In many cases, this is made possible through the communication channels of the system. The transceivers translating messages from the communication bus in the car indicate when functions are unnecessary by issuing a command to go into a Standby state until they are awakened when needed. The transceivers then can relay the message to their respective controller – typically a microcontroller – instructing it to place the system in a low-power state. With more advanced transceivers and system basis chips, multiple functions of this process (transitioning to a low-power state or waking up) can be handled by one device.

TI's CAN and LIN transceivers are designed with all of this in mind, allowing for normal, standby, silent, and sleep modes depending on the device. All of these modes allow the system to be designed in a way that is fully functioning but power-conservative. Power consumption values are available in the data sheets of all TI LIN and CAN transceivers.

## 2 Normal Mode

The normal mode is the fully-functioning mode of the device. The transceiver is in this state when it is being used for communication in its respective electronic control unit (ECU). The transmit and receive circuits are both enabled, allowing a digital input on TXD to be translated to a bus signal and allowing the bus communication to be translated to a digital output on the RXD pin.

This mode is available on all LIN and CAN transceivers and is the highest current-consuming mode due to all of the functions being available.

## 3 Silent Mode

Silent mode is a slightly less current-consuming mode relative to normal mode on some CAN transceivers. This mode disables the transmit driver to the bus, allowing the bus to stay recessive regardless of what is happening on TXD and keeps the receiver fully active, allowing the bus communication to still be observed by the controller/processor. This mode is typically used for applications that need to listen to the bus but not disturb the bus with any kind of communication in response. Silent mode is implemented on the TCAN1051 CAN transceiver, along with several others.

## 4 Standby Mode

Standby mode is a low-power mode implemented on transceivers to give a state consuming less power, but not quite as low as sleep mode. As it is implemented on the TCAN1042, the overall quiescent current from the device is lowered by disabling the bus communication capabilities, and switching to a low-power bus monitor instead. To transition into standby mode, a logic high-level is applied to the STB pin by the controller/processor. This action disables the normal transmit and receive functions of the device, and a low-power monitoring circuit checks the bus for a valid wake-up pattern (WUP) as it is defined by ISO11898-2. Once a valid WUP is received on the bus, the RXD pin toggles low to indicate the mode transition request to the controller/processor. The controller/processor applies a logic low-level on the STB pin. Though the current consumption drops in the transceiver, there is some quiescent current needed to power the MCU so that it can properly monitor RXD for the wake-up indication.

Standby mode, as it is implemented in all of TI's LIN transceivers, acts as a transitional state that the device goes into from sleep mode. When the LIN transceiver is in sleep mode, only the low-power receiver, and some digital logic is left enabled. The RXD pin is floating and the LIN bus is weakly pulled up internally. Once a valid WUP is detected on the LIN bus, the device transitions into standby mode where it is meant to wait for the microcontroller to complete the transition into normal mode. In standby mode, the RXD pin is held low to indicate to the microcontroller that it is time to transition into normal mode, and the internal LIN pull up is re-enabled.

Section 5 explains the two ways that standby mode is implemented, and standby mode as a transitional state

## 5 Sleep Mode

Sleep mode is the lowest power state of a powered transceiver. All bus driving and controlling interface functions are disabled to lower the power consumption of the device, leaving only some digital logic and a low-power bus receiver enabled. Sleep mode also gives a level of system control that is not available with devices that only feature standby mode.

As it is implemented in the TCAN1043 CAN transceiver, sleep mode is entered from normal mode by keeping EN at a logic high and nSTB at a logic low for longer than $t_{GO\text{-}TO\text{-}SLEEP}$. Once in sleep mode, as stated before, all functions except some digital logic and the low-power bus receiver are disabled. The inhibit pin also changes state when in sleep mode compared to normal, standby, and silent modes. The inhibit output is a high-voltage pin internally tied to the VSUP supply in the TCAN1043 transceiver. This output is meant to be used as an enable for a power supply integrated circuit (IC) that supplies the TCAN1043 and controller in the ECU. Once TCAN1043 goes into sleep mode, inhibit goes into a high-impedance state, no longer enabling the power supply, and thus shutting off power to the microcontroller and rest of the system. In this way, the TCAN1043 is used to put the entire system into a low-power mode when it goes into sleep mode.

There are two ways to wake up from sleep mode: a local wake up on the WAKE pin, or a CAN wake-up on the CAN bus. If either of these methods is used correctly, the TCAN1043 switches into standby mode, where the inhibit output is re-enabled. Once this happens, the power supply is re-enabled through inhibit, and the microcontroller is supplied again. The microcontroller then sets the EN and nSTB pins high again, and the TCAN1043 transitions back into normal mode, allowing for full functionality.

The TLIN1441 also has this function, but instead of inhibit, the integrated LDO is disabled in sleep mode, and then re-enabled in standby and normal mode. The process is similar, where the EN pin is set low for 15 µs to go into sleep mode, and all functions on the transceiver are disabled except for some fault-monitoring, digital logic, and a low-power bus receiver to monitor for wake patterns. The integrated LDO also goes low, which is typically used to supply a microcontroller. When a local wake up on the WAKE pin or a LIN wake up happens on the LIN bus, the device transitions into standby mode. The integrated LDO is then re-enabled, thus powering the microcontroller back up, and setting the TLIN1441 back into normal mode through the enable pin from the microcontroller.

In both of these examples, it is shown how the transceiver can be the main control for the system, being the starting point for going into a lower-power state, and also alerting the system when it is needed.

## 6 Conclusion

The low-power modes available on TI's communication transceivers give the system designer options when choosing the correct device. If only lower quiescent current is needed, then standby mode on the TCAN1042, silent mode on the TCAN1051, or even sleep and standby mode on the TLIN1029 is the correct choice. If lower quiescent current and system-level power control is needed, then the sleep mode on TCAN1043 or the TLIN1441 is the better option. Either way, lowering the power consumption without sacrificing time and extra processing steps is made possible through TI's CAN and LIN transceivers. Several transceivers with these modes are listed in this document, but for a full list, visit http://www.ti.com/interface/can-lin/overview.html.

## IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

# Semtech LoRa Transceivers – a KISS approach to Long Range Data Telemetry

Stuart Robinson
January 2015

## Introducing LoRa

In 2013 Semtech released a new RF IC transceiver designed to be used for the industrial scientific and medical (ISM) market, this was the SX1278 and its variants. These transceivers use a proprietary spread spectrum transmission method they called LoRa; short for Long Range. The claim is the devices are capable of receiving signals at below noise level, the limit being -20dB signal to noise ratio (SNR) at a spreading factor of 12. Frequency shift keying (FSK) transceivers such as the RFM22B need a positive SNR of somewhere between +5 to +10db, so potentially the LoRa device can operate with signals that are up to 30dB weaker than those required by the RFM22B and similar FSK devices.

A 30dB improvement in link margin is an approximate range gain of 30 times so the LoRa devices do look just a little bit promising.

The Semtech SX1278 was incorporated in 2014 in two ready to use UHF\434Mhz modules, the Dorji DRF1278F and the Hope RFM98. This report will investigate the performance of these 434Mhz modules.

## Experiments with FSK Transceivers

The Hope RFM22B is an example of a low cost FSK transceiver module that was able to provide a communications link for two low Earth orbit amateur satellites, $50SAT and T-LogoQube. The results produced by the T-LogoQube team being particularly impressive as they demonstrated 2 way communications at up to 2,700km with only 100mW output from the satellite.

Such long distance operation of the FSK transceiver modules required the use of a 50W power amplifier for uplink, a low noise amplifier (LNA) for the downlink and an antenna farm of accurately tracked twin 20dB long yagi arrays to follow the satellites path across the sky. Whilst it worked well the equipment is not simple to set up or operate and is expensive.

The $50SAT team, Howie DeFelice, Michael Kirkhart and Stuart Robinson were able to receive the data telemetry at up to 1200km from $50SAT using hand held antennas and low noise amplifiers.

These ISM band transceivers have up to 100mW output although for license exempt ISM use the limit is normally 10mW. For amateur band applications amplifying the power from these simple 434Mhz transmitters is not that difficult. Mitsubishi do a series of UHF power amplifiers that will take a low input power, 25mW or less, and boost it to as much as 60W.

FSK transceiver modules do need a good LNA in front of the receiver for long distance operation, the LNA adding around 12dB of useful signal gain, which is a range improvement of about 4. To use the LNA and power amplifier together you need either separate antennas for TX and RX, (not very convenient) or sequencing and switching of the single antenna between RX and TX modes. This is all possible of course, but is not simple or cheap to set-up.

The $50SAT satellite side installation was an example of a 'keep it simple' system, but this was clearly not the case for the ground station set-up. To reduce complexity and cost we need to eliminate the LNA, power amplifier, output filters, sequencer, antenna farm, rotators and connect our humble transceiver module direct to a single simple antenna, but how far are we going to get at only 10mW or 100mW with FSK transceivers?

Assuming ¼ wave antennas at both ends of a link an FSK transceiver will at 1000bps go around 12km at 10mW and 40km at 100mW. 12km for 10mW or 40km for 100mW is good, but it does not allow for a low complexity set-up for satellites or high altitude balloon use where ranges of 200 to 1000km may be needed.

FSK transceiver modules can be used for long range *reception* however, by using a low cost LNA and or high gain antennas. In mid 2014 I launched a Pico balloon using an RFM22B as the transmitter. With a Diamond X50N vertical antenna (5dB low horizon gain) and £50 LNA kit I received 1000bps telemetry from the balloon tracker with 10mW output at 174km. Since balloons don't move around the sky much, a long yagi could be set-up on a mast or tripod and it would not be difficult to manually point it towards the balloon. However, the uplink would be troublesome, since 10mW into the yagi would exceed the effective radiated power (ERP) limit of 10mW.

Although this is a report about the LoRa transceivers, I do make many references to the RFM22B. I did a great deal of testing with this module for the $50SAT project and I can therefore make direct comparisons, by using the same tests in the same locations, between an FSK transceiver and a LoRa one.

## Testing LoRa

The first tests I carried out on the LoRa devices were on the Dorji DRF1278F module. There was some working program code available for Arduino, so I connected the DRF1278F to a 3.3V modified Arduino UNO. The tests described were carried out at UHF frequencies at around 434Mhz.

## The Long Walk.

I don't, unfortunately, have the benefit of a laboratory full of test equipment or a nearby large underground cavern far from sources of radio interference and noise, so I make do with the environment around me in South Wales instead.

In practice these transceivers will be used in real world situations, so it seems appropriate to test them in similar situations. A good laboratory performance is one thing, but a ground based receiver will probably be in the middle of an urban or industrial area and have to cope with all the associated RF Noise, so why not test it there?.

The Arduino program code was changed to put out a data transmission at approximately 60bps at 10mW, the low data rate chosen to maximise range. An RFM22B was then programmed to send data at 1000bps at the same power. The data rates were not the same, but testing for $50SAT showed that the RFM22B performance did not improve much at data rates below 1000bps anyway.

Both transmitters were mounted in similar tin boxes from the local £1 shop, with a 20dB attenuator in-line with the transmit antenna. The actual radiated power from the antenna was therefore only 100uW. Each transmitter had a matching receiver that made an easily heard beep whenever a packet was received

Comparing the performance of the LoRa device to the RFM22B meant that if a dB performance relationship could be established between the two devices, we could extrapolate from the known performance results of the RFM22Bs that have been in low Earth orbit and then form a view as to the distance performance of the LoRa device in the same situation.

At the beginning of September 2014 I was staying in the South Wales seaside resort of Tenby and the beach provided a good testing range mostly free from obstructions (at low tide) see pictures below.

The testing principle here is simple, walk away from the transmitter until the beeps stop. The distance difference can then be converted into a dB performance ratio.

With the RFM22Bs the reception stopped at approx. 100M.

Rather a longer walk was needed with the LoRa devices, approx. 1km, so 10 times further. 10 times further represents a link improvement from the LoRa device of 20dB over the RFM22B.

I was very surprised by these results and impressed, but a method was required to allow more accurate comparisons between the LoRa device and the RFM22B without the need to walk very long distances.

## Descending Power Testing

With both the LoRa Device and the RFM22B you can vary the output power of transmissions. If you set up the devices so that the transmitter sends a series of descending power packets, with the power level sent as part of the packet, the receiver can record at which power level reception stops.

First set up the link with the transmitter and receiver some distance apart. Set up the transmitter and receiver to a particular data rate and you note that in this case reception stops at 10dBm from the transmitter, i.e. the 11dBm packet is received but the 10dBm and lower power packets are not. You change data rate settings to a lower data rate and note that reception now stops at 5dBm. You can conclude from this that the lower data rate has a 5dB performance benefit over the higher rate. A better performance for the lower data rate would be expected, but you have now quantified the difference and can convert this into a distance improvement. A 5dB link performance improvement would increase distance by 75%.

The typical output of the receiver program is as below;

Powers,16,16,16,15,16,16,16,16,8,2,0,0,0,0,0,0

With 17dBm on the left, and 1dBm per step you can see performance is consistent in this test down to 10dBm, only half the packets received at 9dBm, and only 2 at 8dBm.

I next set-up a LoRa transmitter and a RFM22B transmitter to send out data packets at 1000bps at a location approximately 750M away from my home workshop across an urban area. The transmitters were side by side on a short pole about 1M from the ground, and on separate frequencies of course.

I adjusted the height of the antenna (a QF Helix) on my mast so that I was only getting the most powerful 17dBm packets from the RFM22B, the lower power packets were not being received. I then swapped across to the Dorji DRF1278 receiver, and I was also only getting the most powerful 17dBm packets.

This does not seem to be any improvement, apart from the fact that the antenna on the remote LoRa transmitter was fitted with an in-line 20dB attenuator!!!

The software also turned on the transmitter for a brief period with no modulation and at each power level. When received with a Funcube dongle and SDR# software, this signal appears as a spike on the SDR# display, so you can measure the signals received dB level. In this way I could confirm that the LoRa transmission was indeed 20dB lower than that of the RFM22B.

So in this test, across a typical urban environment, the LoRa device has a link budget gain of around 20dB over the RFM22B at the same data rate, which equates approximately to a range difference of 10 times.

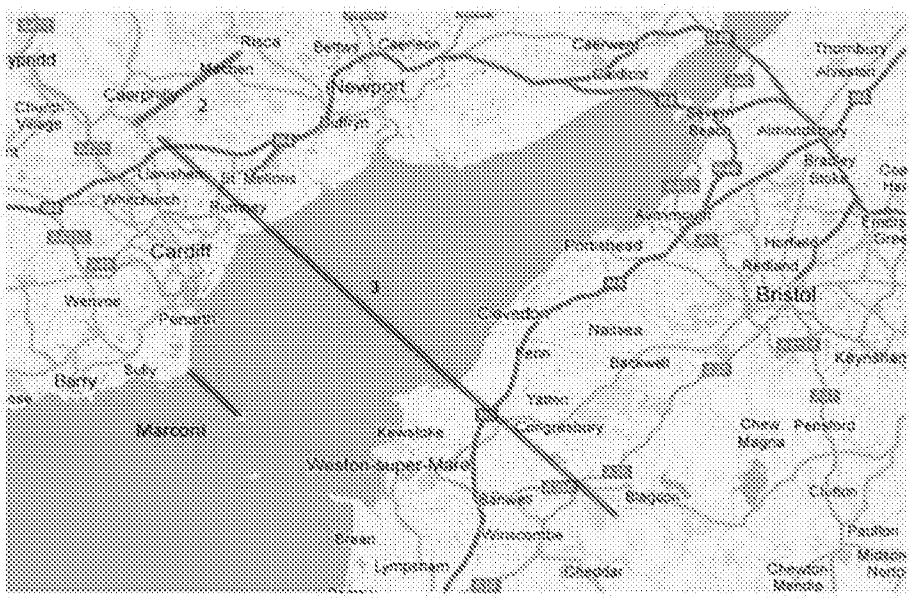
I re-programmed the LoRa device so that it was transmitting at the lower data rate of 100bps. The LoRa reception now stopped a further 10dB lower, the 7dBm packets were now received. Thus the 100bps LoRa

transmission has a potential range improvement over the 1kbps RFM22B transmissions of 30dBm (20dBm + 10dBm) this represents an approximate distance improvement of 31 times.
So if I needed 17dBm to cover 750M in an urban environment then the lower power of 10dBm\10mW (the ISM limit) would have a range of about 350M, so how far would the LoRa device go? The tests suggest LoRa would manage 3.5km at 1kbps and 11km at 100bps!

## The Long Drive

I did the same long range line of sight test that I did for the RFM22B back in 2012, 40km between two hilltops across the Bristol Channel (route 3 on the map below). This original test was for the $50SAT project, some figures for real world performance were needed to do uplink and downlink budget calculations.

Although the distance was 'only' 40km it was a fairly long drive to get from Cardiff into position on a hilltop across the Bristol Channel.




Route 'Marconi' on the map is the where the first radio transmission across water took place in 1897, by Mr Marconi, not sure of his call sign.

The route chosen had good line of sight and had plenty of altitude in the middle of the route to minimise Fresnel zone effects, see the profile below.


The RFM22B had been sending data at 1000bps in the original $50SAT test. The LoRa device was programmed for 1042bps (Bandwidth of 41.7 kHz, Spreading factor of 8, Coding Rate of 4:5).

Antennas were simple 1/4 wave wires at both ends, transmitter and receiver were in tin metal boxes with no real ground plane, as for the previous tests mentioned in this report.

At the receiver end all that was necessary was to listen with a UHF hand held for an audio tone, this indicated the start of the 17dBm to 2dBm sequence and count the beeps, 1dBm less for each beep. The terminal output of the receiver was also saved as a log file for analysis. Reliable packet reception for the RFM22B had needed 20dBm for the 40km, the LoRa device only 3dBm, so a 17dB performance gain.

**17dB is a significant improvement!**

Following the 40km\2mW LoRa result, Howie DeFelice produced a link budget calculation spreadsheet for the LoRa devices.

One anomaly that was clear from the results of Howie's link budget calculation was that the signal strength of a 2mW transmitter at 40km calculated as -114dBm.

As 3dBm\2mW was the proven limit of reception, you can conclude that the receiver needed at least -114dBm of signal to reliably operate. However, the Semtech LoRa calculator application claims the sensitivity at the bandwidth and spreading factor used was -131dBm, so where has the missing 17dBm gone?

The sensitivity quoted in the data sheets is probably a figure that cannot be achieved, at least in non-laboratory situations on Earth, since the real life background noise level is so high. I checked and a typical background noise level reported by the SX1278 RSSI register during the 40km test was around -100 to -105dBm. My RF Explorer spectrum analyser reports a similar level. With noise at that level there may be little value to be gained by the LoRa receiver having a sensitivity of up to -148dBm as the data sheet specifies.

Where the LoRa device seems to be is getting it's real world performance from (it is clearly substantially better than the FSK receiver in the RFM22B) is its ability to receive signals below the noise level. The acceptable signal to noise ratio (SNR) for the spreading factor used in the 40km test (8) is quoted in the Semtech data sheets as -10dB, so the receiver should work if the signal was 10dB below the received noise level. 10dB below a noise level of -100dBm is -110dBm, close to the predicted signal level that 40km\2mw would have produced (-114dBm).

So whilst the receiver predicted sensitivity (of -131dBm) might suggest far greater range than 2mW for 40km gives in reality, it's the relationship between local noise level and SNR performance that appears to predominate.

I then carried out some shorter range tests using the same descending power test method as above, but with the antenna output fitted with an attenuator in order to cut the distance down. At a spreading factor of 12, the data rate drops to circa 100bps, and signals were received at 10dBm lower, the SNR at this point is reported as -20dB.

So reducing the data rate from 1000bps (used for the 40km test) down to 100bps, does appear to increase link budget by a further 10dB, reducing the data rate to 50bps gives 13dB link gain.

## Problems?

There was one issue noted in the initial testing of the hardware that needed further investigation.

I was testing at a bandwidth of 20.8 kHz, Spreading factor of 11, this gives around 100bps and would be a good replacement for the FSK RTTY often used to send tracking information from a high altitude balloon in the UK. In the UK amateur radio license holders cannot transmit from 'airborne' devices so we are limited to 10mW ISM stuff.

It appears that there was something happening with the higher power packets at the 17dBm to 14dBm level. In the receiver report below the higher power packets were received but had CRC errors. The weaker packets are received fine.

Powers,0,0,0,2,5,5,5,5,5,5,5,5,5,5,5,5

The packets were well attenuated at the transmitter so it seems unlikely that the transmitter was over driving the LoRa receiver.

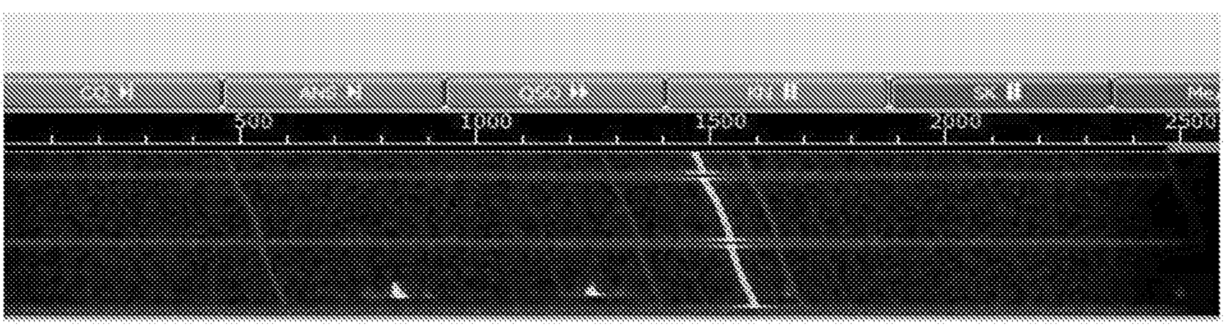
The cause of the problem appears to be the same heating that affects the RFM22B\Si4432. The IC package heats up when the transmitter turns on and this alters the oscillator frequency slightly since the oscillator circuit itself heats up as the chip heats up. This shifts the frequency and is very easy to spot when receiving a signal in FSK side-band mode

It may be that just replacing the crystal with a plain oscillator module would be enough to improve the short term heating issue, it would take a while for the TX heating the chip to cause a separate oscillator to heat also. To be fair the data sheet does suggest that the device is susceptible to short term stability issues, so the issue should not have been a surprise.

I decided to measure the frequency drift of the RFM98 when the transmit was turned on. The transmit was turned on from cold (actually room temperature) at approx 434Mhz with a 10mW carrier and the change in frequency measured at intervals. I used a Funcube Dongle to receive the 434Mhz signal in LSB mode, and the FLDIGI software to display the resultant audio tone; this was the resulting drift away from centre frequency at 5 second intervals;

5 – 50Hz
10 – 120Hz
15 – 160Hz
20 – 210Hz
25 – 240Hz
30 – 260Hz

After one minute the drift was down to around 10hz every 5 seconds. It looks like this on screen, the gaps in carrier are at 5 second intervals, the scale marked 500,1000 etc. is in hertz.



The problem (packets received with CRC errors) did not seem to occur with high data rate (1042bps) packets, but was present on the lower data rate (98bps) packets. It was also noted that the errors did not occur in short packets of 10 bytes or so, but did with longer ones. The problem also seemed worse with lower bandwidth packets.
There would appear to be a limit for the allowable frequency drift from the start of the packet which is a small percentage of the bandwidth. The frequency tolerance is stated at 25% of the bandwidth, so at 41.7khz, that's 10.4khz and the short term drift apparently causing the problem is nowhere near this, as measured above.

What could be happening is that at the preamble time the receiver locks on to the incoming frequency and decodes appropriately. If the transmitted signal drifts by a small amount (I suspect its in the order of 100hz) the receiver decode fails. So lower bandwidth and lower data rates meaning longer packets would be more susceptible to the problem.

The immediate response is no doubt to suggest that the crystal on the Hope or Dorji modules is replaced with a TCXO or maybe just a plain TXO, so that the oscillator circuit doe not get affected by the TX heating effect. But is such a modification really necessary?

It was pointed out by someone on the UKHAS groups that there is a low data rate optimisation bit that I had omitted to set, and this does indeed reduce the problem significantly, especially at a bandwidth of 40.7khz. However this is a potential issue to be aware of, possibly in satellite applications, where the rate of change of centre frequency during the actual reception of a packet can be significant.

There is also the issue of the differences in frequencies between a pair of modules.

## Frequency Offsets

The LoRa devices do not have an automatic frequency control circuit (AFC) as such and if the transmitter and receiver differ in frequency so that they are outside of the frequency tolerance for the bandwidth used, the link just stops working, there is no gradual fall off in performance, it just fails.

The AFC in the RFM22B is very effective, there is very little degradation in performance when transmitter and receiver were offset by up to 10khz, and this meant no correction was needed to cope with the Doppler shift from a satellite in low Earth orbit.

I have several RFM98s and DRF1278Fs here, and the crystals do vary in base frequency. If the modules are programmed with the same frequency settings and bandwidth of 20.8 kHz, two of these modules just don't communicate, the crystals are far enough apart in base frequency to prevent the link working and at frequency separations that are less than the data sheet would suggest. One module was 2.1khz high the other 1.2khz low (at the same temperature). These crystals were 3.6khz apart, the failure point should have been 5.2khz..

You can of course adjust the programming of the SX1278 to 'calibrate' the crystals by making fine adjustments (in 61 Hz steps) to bring the crystals base frequency together.

The SX1278 data sheet specifies the frequency tolerance as +/- 25% of the bandwidth, so the 41.7 kHz bandwidth should have a frequency tolerance of +/- 10.4 kHz, so within the Doppler shift range for a low Earth orbit satellite. Higher bandwidths would improve frequency tolerance further, but the higher bandwidth will introduce more noise into the receiver, and reduce sensitivity.

In the UK, if you want to use the ISM bands for continuous duty applications, then the channel use needs to be 25khz or less. At a bandwidth of 20.8khz, its very likely that a TCXO is needed, but at 40.7khz you can manage without, but you do need to keep the on-air time under 10%, which is not a problem for many applications.

As my quest was to keep the long distance link set-up as simple as possible, I decided to stick to a bandwidth of 40.7khz, as it appeared retro fitting a TCXO to the modules would not be required.

## So what does this initial testing tell us?

It's unclear if the LoRa device is intended for long duty cycle operation of 20dBm (100mW) but it is at 17dBm (50mW). So based on the above tests what sort of range could we expect with the LoRa devices running at 50mW, with just simple ¼ wires as antennas?

**Estimated range at 1000bps, 50mW = 200km**
**Estimated range at 100bps, 50mW = 633km**

The antennas used in the tests above were fairly crude and without an effective ground plane, so it would be reasonable to assume that actual distance, ground to satellite (or balloon) achieved would be even further. This did prove to be the case as a subsequent test revealed, see later in this report.

Also note that an easy build DIY vertical gain antenna, such as a Slim Jim, at the ground station would extend the 633km to close to 1000km.

This report has concentrated on the performance of the LoRa devices without any additional amplifiers in line with the keep it simple approach. The LoRa devices are easy to program for higher data rates which could possibly be utilised if the ground station has better equipment, such as low noise amplifiers and high gain and\or tracking antennas. However the addition of such equipment considerably increases complexity and cost, and the initial testing does suggest it is just not needed.

It would be feasible to make a long range communications system adaptive, send simple commands to a LoRa device at a very low data rate (easy to receive and long range) and have the link switch to much higher data rates when you have the antenna systems that have the gain that will allow the higher data rates to work.

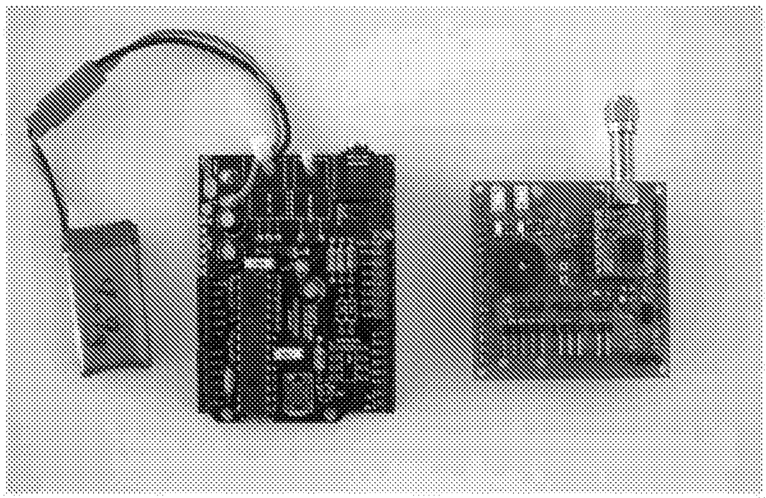
## Long Range Line of Sight Testing.

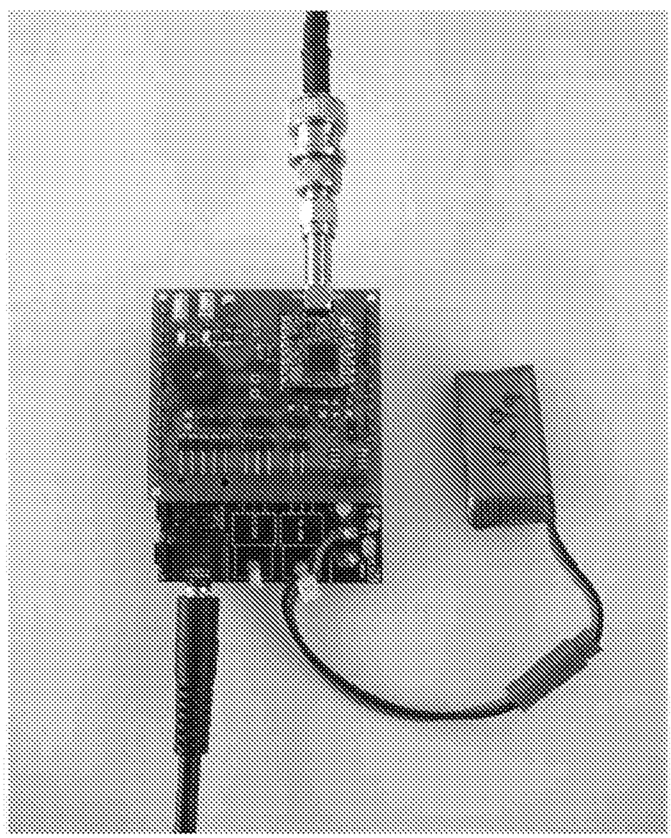
Next I considered the problem of very long distance line of sight (LOS) tests. It will be necessary to have one end of the link at fairly high altitude since the curvature of the Earth means that even for high mountain tops a receiver may be below the radio horizon from the transmitter.

A low cost way of performing LOS testing is to build a small GPS balloon tracker that reports its position back to you. These trackers can be launched using 36" foil party balloons filled with helium. If weighted correctly they will rise to around 6000M to 8000M and float along at that altitude

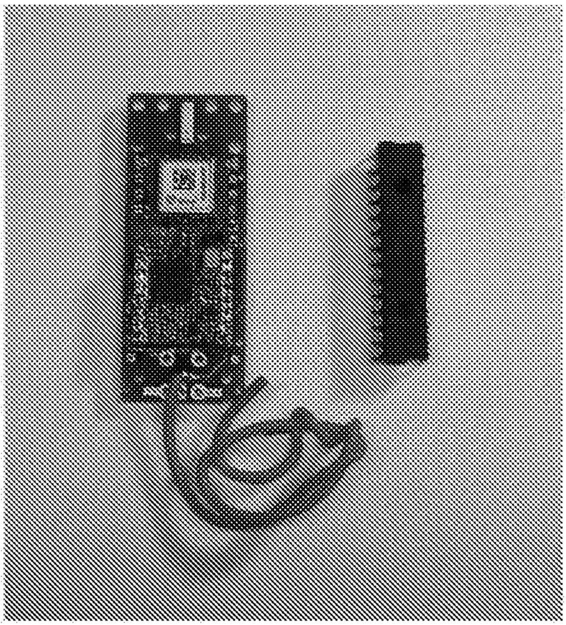
I first produced an Arduino Shield for the RFM98 LoRa device. The shield was designed primarily to allow easy development of the LoRa GPS tracker software, but it can be used as a tracker transmitter in its own right and is also intended to act as the LoRa base station receiver.

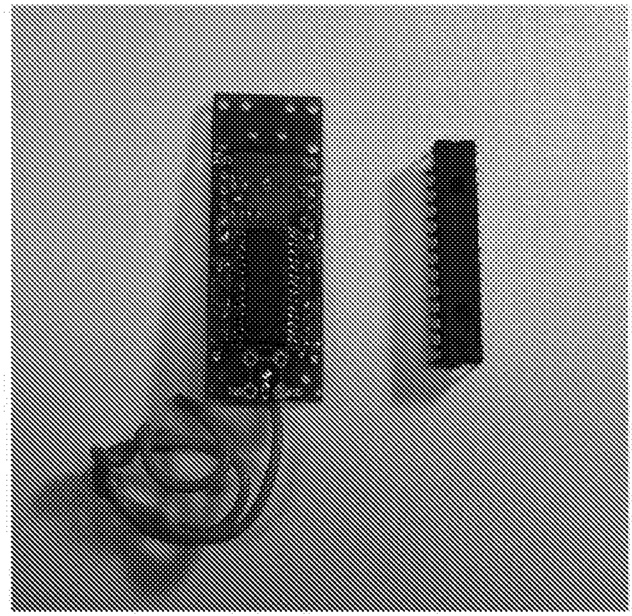
This shield is currently running under PICAXE code on the appropriate PICAXE shield base, but the same shield (see pictures below) should work with an Arduino UNO (that's been converted to 3.3V use) as well.





The balloon would need a small tracker, under 20g including battery, so I designed HABAXE2, see bellow;





HABAXE2 used a PICAXE 28X2 and the Hope RFM98 radio transceiver. This Pico balloon tracker weighed around 16g with a 10g 380mahr Lithium Polymer battery and was attached to the 36" foil party balloon, filled with helium and adjusted for 2g of free lift, see picture below.

The supplier I use for my PCBs, including the one that has now been in low Earth orbit for 14 months and still working, charges $14 for 10 boards up to 50x50mm including worldwide postage.

The foil balloons are made by Qualatex. The headland in the background on the horizon is Lavernock point and behind it almost obscured by early morning mist is Flatholm island, the site of Marconi's first transmission across water in 1897.

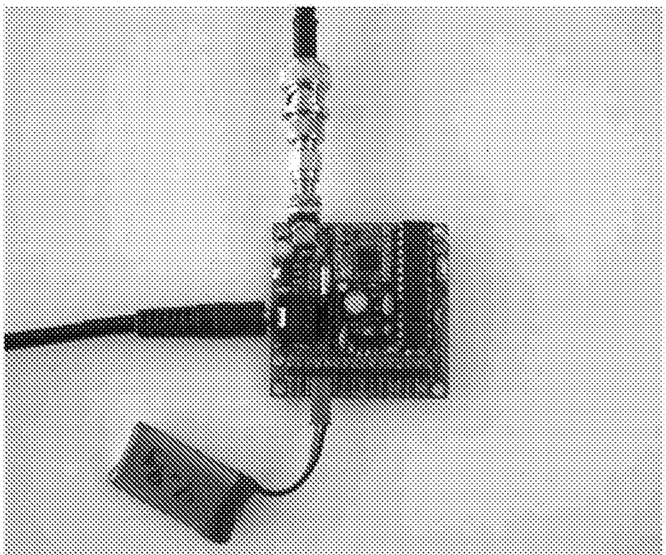
The RFM98 radio transceiver was sending the UKHAS compatible tracker payload as 100baud FSK RTTY and the same data as LoRa digital telemetry. The antenna was a vertical ¼ wire with 4 x ¼ radials, at 434Mhz.

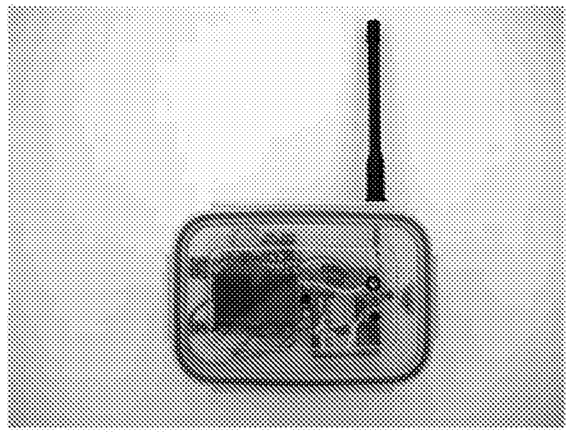
Radio amateurs would be listening in for the 100baud FSK RTTY signals from the Balloon, unfortunately there are not many LoRa listening stations. Software on their PCs translates the signals and then posts the information via the internet in real time to the central Habitat\Spacenear tracking system. This displays the balloons position and other information in the data payload such as battery voltages and temperature.

One objective of this first flight of HABAXE2 was to see if the RFM98 device could be used for HAB tracking using the LoRa digital telemetry. To this end the ground station receiver was set-up to receive the LoRa tracking data (position) from the balloon and convert it back into NMEA GPS sentences. This was then fed back into a PC via a serial connection so that a mapping application could display the track of the balloon on the local PC screen. I could therefore self track the balloon, but would obviously loose its location once I could no longer receive the LoRa at my location. This concept was also tested using the portable LoRa receiver described elsewhere, that was connected to a small netbook PC and I tracked the first 30 minutes of the flight on a map screen, no Internet connection was needed.

I had already established in the earlier tests that the LoRa telemetry at 1042bps could cover 40km line of sight (LOS) with only 3dBm\2mW and a simple ¼ wave wires with no radials, so it was clear that more then 100km was expected at the full 10mW and a better matched antenna.

Part of my overall objective was making a LoRa base station that was as simple as possible and to also build a portable receiver. A simple PCB was developed using DIP PICAXE 28X2, this was used both as a base station receiver and for a portable receiver (described in more detail later).





The base station antenna I used was an omni Diamond X50N co-linear, on a mast attached to my shed (I live in the middle of a city). The antenna has a forward gain close to the horizon of around 5dB, similar to magnetic mount antennas for cars.

That is all there is to the receiver hardware and it will run for 12 hours or more from a single Lithium Ion camera\phone battery.

The balloon was launched early one Sunday morning and I retired to the local Café for breakfast, listening for the tracker on the portable receiver I had built.

All looked OK, tracker battery, GPS and the balloon climbing. The track was curling round to the South and East as predicted by the CUSF flight predictor so after finishing breakfast I went home to track from there. The balloon climbed to about 8000M and continued to float along at that altitude.

One of my objectives was to ensure the tracking could be carried out in unattended mode so at home I had left the LoRa base station receiver running. This receiver was set to receive the LoRa tracker packet and upload it into HABITAT with an AFSK RTTY transmission picked up by my Funcube dongle.
The first packet was received whilst I was still in the Café 5 miles away and it went into HABITAT OK;

**$$HABAXE2,127,09:37:05,51.5372,-3.2062,87,62,357,0,0,0,6,4110,13,Y,0*079A**

The payload starts with the station identifier (HABAXE2) then continues with the sequence number of the transmission (127) the GPS time, Latitude, Longitude, Altitude, speed, track, some internal parameters, the battery voltage (4110mV) etc.

A two way command uplink capability was implemented for HABAXE2, the tracker would listen for incoming commands (as LoRa telemetry) for approx 5 seconds just after sending the FSK RTTY. These commands could be used to make the tracker perform various actions, such as warm reset, send RAM variable information or flags for remote diagnostics, send particular NMEA sentences or send a series of descending power packets at a particular set of LoRa modem parameters.

It's the ability to send descending power packets that would form the basis of the LoRa capability tests, this test method was described earlier. If you commanded the tracker to send a sequence of packets from 10dBm down to 2dBm, with a particular set of LoRa modem parameters and flipped the base station across to listen with the same set of LoRa parameters, you could measure the effectiveness (and hence distance capability) of the particular mode.

The SX1278 needs very few register changes for different LoRa modes and assuming that explicit header mode and CRC mode were on, the payload which is the command to be sent to the remote tracker, possibly hundreds of km away to make it reply with a set of test packets only needed to contain these bytes;

RegModemConfig1, RegModemConfig2, RegModemConfig3, startdBm, endBm, iterations.

The actual payload for ever packet was preceded by 3 bytes, the packet type (telling the receiver what to do or how to act on the data) the destination station (0-255) and the source station (0 – 254). A destination of 255 was assumed to be a broadcast, so 255 stations could potentially share the same channel.

On receipt of this test request, packet type 43, the remote tracker would recognise it, program the LoRa modem accordingly and reply with a sequence of packets starting at a particular dBm power and ending at a particular dBm power and do this is a number of times (iterations).

The ground station would (on receipt of an acknowledge packet from the tracker) flip across to the new LoRa settings and listen for the replies. The reply packets (packet type 44) would be seen by the ground station and would extract from the packet payload the power level used for transmitting the packet. Part of the packet type 44 payload was the power level used. The ground station would display the power level on a serial terminal screen together with received SNR and RSSI information for later analysis. So when the packets stopped coming in, you knew the limit of reception for those LoRa settings, and could extrapolate upwards to what distance could be expected at the 10dBm\10mW ISM limit or the full 17dBm\50mW for other applications.

Thus for one test, I used the higher data rates, bandwidth of 500Khz (BW500Khz) spreading factor 7 (SF7) and a coding rate of 4:8 (CR4:8) that gave an equivalent data rate of 13.7kbps according to the LoRa calculator. At my base station I received the above packets down to 7dBm at 105km, the distance was taken from the location in the previous tracker payload. Thus you could conclude that this 13.7kbps rate would have a range at 10mW of 150km.

The main tracker payload and 2 way communications were being done at BW40.7Khz, SF8 and CR4:5, approx 1042bps, so this would have been expected to operate over a longer distance than the 13.7kbps telemetry.

Putting a good LNA in the receiver side does help with the low spreading factor rates (lower allowable signal to noise Ratio (SNR)) and in this case my masthead LNA allowed me to receive

the same 13.7kbps packets at 2dBm, so an 8dBm margin. Thus at 10dBm it would be reasonable to assume a potential range improvement of 2.5 or a distance of around 250km for the 13.7kbps rate.

The main tracker payload (same as the FSK RTTY) was being sent out as LoRa at the 1042bps rate. The last time this main tracker payload was received error free was at a distance of 269km, and note no LNA was used.

Once the high rate 1042bps packets were out of range at 'only' 269km I could not test communications any further since the command link was also set at 1042bps. However at 242km I did command HABAXE2 to send a series of 98bps SF12 packets which were received error free down to 2dBm\3mW. Extrapolating the 2dBm upwards to 10mw (UK limit) would represent a UK legal range @ 10mW of 611km, which is the radio horizon at an altitude of circa 22km.

There was a problem with the low data rate 98bps packets, due to an omission on my part that the low data rate optimisation bit was not set for the trackers LoRa modem. This did not affect the shorter test packets but the short tracker payload of 22bytes being sent at @ 98bps was being received with CRC errors. For anyone listening out for the 98bps during the flight, if you received it (but got a CRC error) bear in mind that the TX power from HABAXE2 had been set to 2mW only. So at whatever range you received the 98bps during the HABAXE2 flight, it would likely have been received at approx 2.5 times that range assuming the tracker had been set up correctly with the low data rate optimisation bit set and the full 10mW had been used.

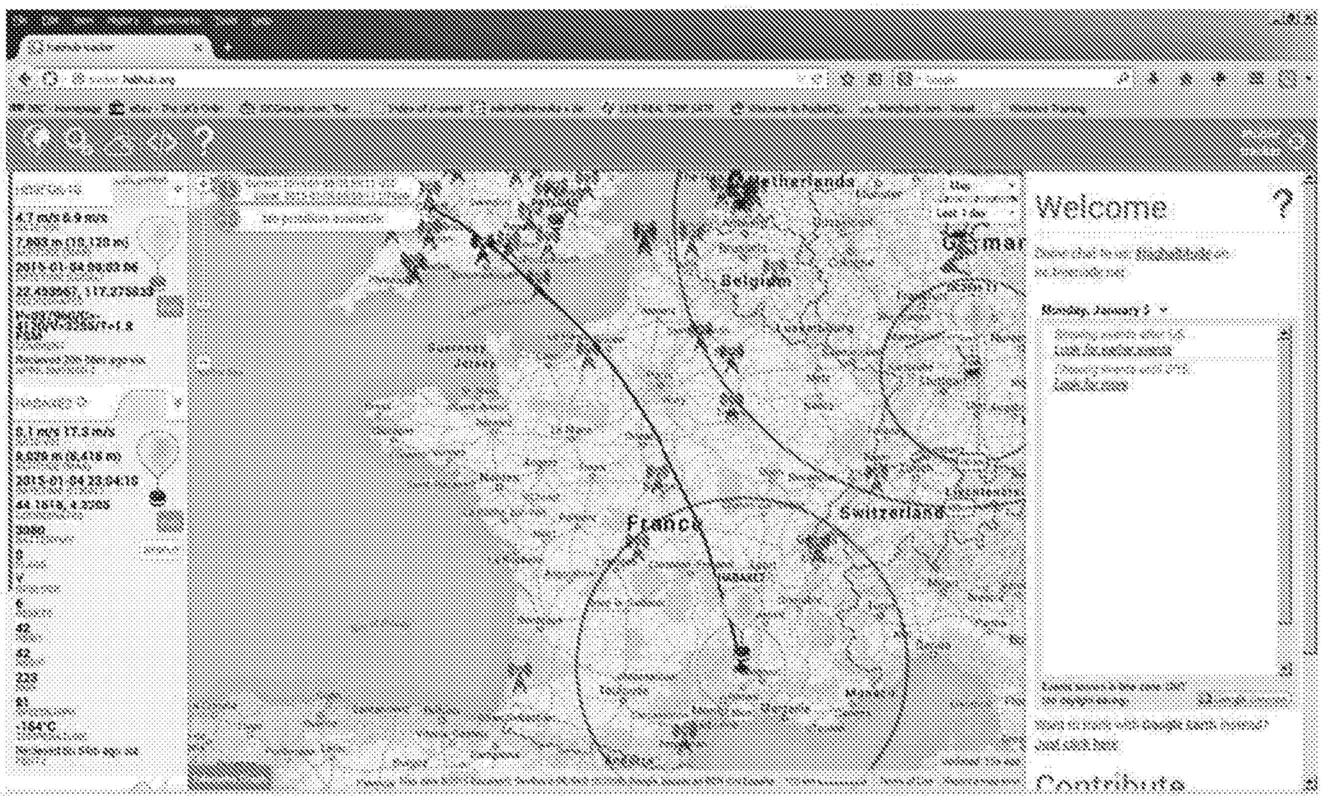
I also implemented and tested a method of adjusting the base station frequency to match the tracker frequency which could drift due to the cold. This was done by switching the base station antenna to the Funcube\SDR# and using the SDR# screens frequency cursor to mark the incoming FSK RTTY idle frequency. With the antenna then switched back to the LoRa receiver a menu option allowed a small marker carrier to be sent out which could be shifted up and down via the keyboard to match the frequency cursor on the SDR# screen. This method worked and would be required to allow the LoRa transmissions below 41.7khz to be used with the standard RFM98.

## Flight Details

The balloon was launched around 09:30 on the 4th January 2015, from Caerphilly Common, 51.5621N 3.2228W. It was last heard of at Latitude 44.1618N, Longitude 4.3205E at an altitude of 8032M, having travelled just over 1000km.

Due to an error the temperature was being displayed at an offset of minus 128C, so the last recorded temperature was actually minus 36C.

This is the track as displayed on the Spacenear\Habitat system;




These days 1000km for a Pico balloon is not very far at all, at the end of 2014 one Pico balloon completed 6 times around the Earth!

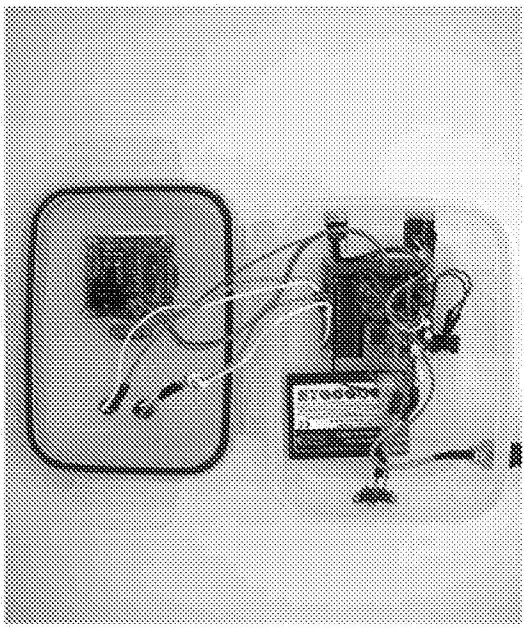
HABAXE2 had no solar panels fitted, so it was designed to last only 24 hours or so.

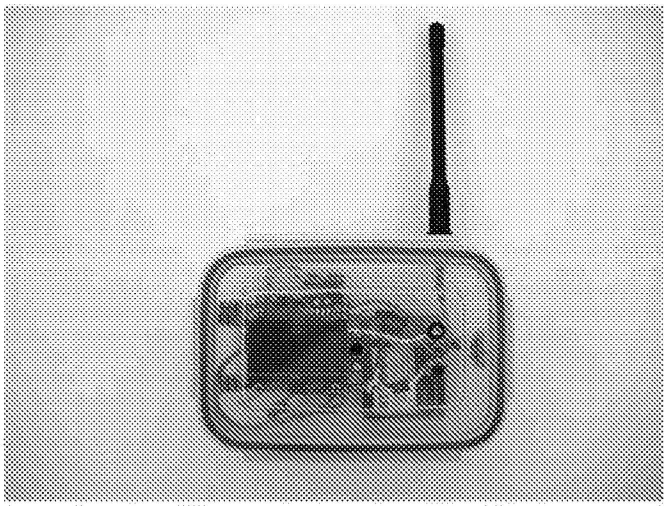
## Portable LoRa Receiver

One advantage of using telemetry to track high altitude balloons (HABs) is that it's possible to build small battery powered receivers with simple electronics to display location information from the tracker in flight or one that has landed and needs to be found.

High Altitude Balloon trackers in the UK normally use FSK RTTY which does have very long range capability, hundreds of km. Whilst this can be received with portable equipment, the minimum set-up is probably a reasonably powerful notebook PC and a Funcube dongle which needs constant attention to tuning due to transmitter frequency drift.

In comparison a telemetry receiver can be as simple as a RFM98 based board, with suitable software, running from a single Lithium Ion battery and driving a serial LCD display. Total parts cost around £25. The LCD can display the received tracker information such as Latitude, Longitude, altitude, temperature etc.





I have found the best displays for this type of application to be the small Digole serial 128×64 graphic LCDs. These are low power and some versions will run from supplies that can vary from 3v3 to 5V with no contrast adjustment needed as the supply voltage changes. These serial LCDs require only a single output pin from a micro controller to drive them. The 128×64 displays will show 7 lines of 21 characters at font size 10, there are other font options and graphics too.

The base station receiver board for the RFM98 and PICAXE 28X2 processor was used for the portable receiver, it is 50mmx50mm. The PCB has connections for the serial LCD, LED, buzzer and switch. The board also incorporates a single cell lithium ion charger IC, so all that's needed to charge the battery is to connect the external power input to 5V from a USB connection.

The hi-tech case is a clip top food container. These are cheap, robust and easy to work. You don't even have to cut a hole for the display. For the antenna connection I use a short SMA plug to chassis socket pigtail. A bit of hot melt glue here and there secures the bits in place.

## Other uses for a portable telemetry receiver?

One of the worst cases scenarios for receiving the telemetry is if the tracker (transmitter) is on the ground (lost?) in an urban area as the buildings will cause heavy attenuation of the UHF signals.

The RFM22B telemetry in the urban area around my home when a transmitter was on the ground and using a hand-held receiver had a range of around 80m maximum @ 10mW, not a lot of help in locating something 'lost'.

If the 98bps LoRa telemetry contained the Latitude and Longitude, and assuming this GPS derived location is correct, how far away can it be received on a portable receiver?

Two options were tested, first just the portable receiver with its rubber duck style antenna and the same receiver connected to a magnetic mount 4dB gain vertical antenna on a car.

The LoRa tracker transmitter was placed with the antenna (a 1/4 wave wire) horizontal on the ground in my garden. I then went for a walk up to see how far away the tracker transmissions were received. reception stopped at 825m, not bad at all.

How far away would the tracker be received by a 'chase' car with my modest magnetic mount antenna? I left the tracker running and drove to work. I last got the payload from the tracker when I was 2.2km away, a considerable improvement and clearly viable as a simple means of locating a landed but lost balloon, assuming the GPS location being transmitted is valid of course.

For those that believe a PICAXE cannot be used for serious tasks, then think again. PICAXEs are simple to use for beginners and skilled programmers can make them do serious tasks. There are other platforms for these type of applications of course which some may choose to use

## What Next?

HABAXE2 used the higher LoRa rate of 1042bps for the command and control link. Since the commands are generally short they can be switched to the lower data rate of 98bps which would provide for a much greater command and control distance. It would then be possible to use a high gain antenna at the base station and thus a high data rate for receiving data and so then reduce the power for the uplink commands and still stay within ISM band ERP limits.

Would a simple modification of the RFM98 modules allow a TCXO to be used? If it were simple then the lower bandwidth of 20.8khz or lower is feasible, which would then allow high duty rate use above 10%. However, if a low duty rate (<10%) is acceptable, then you can use a bandwidth of 41.7khz, where performance seems to be acceptable with an unmodified RFM98.

There is a cut away option for the Pico tracker to be flight tested, instigated by an uplink command. The cut away uses a small LiPo and Nichrome cord to cut the suspension strings (low melting point dyneema), tests on the bench indicate it should work.

For very long distance testing of LoRa it's going to be necessary to put the LoRa transceiver on a low Earth orbit satellite, very likely a PocketQube like $50SAT. There are some issues with Doppler to be investigated, but if they can be overcome, it's possible the LoRa will allow telemetry to be received from a low Earth orbit satellite (albeit at a low data rate) with a very low cost portable receiver such as that described above. I don't think this would have been possible before the clever people at Semtech came up with LoRa.

And finally, many thanks to those at UKHAS and HAB enthusiasts in France who assisted in tracking of the HABAXE2 flight.

**Stuart Robinson**
**GW7HPW**
**January 2015**